

Kartkówka 30 IX, grupy zmieszane

0. (2+2 p.) Rozwiń skrót i napisz, jak nazywa się po polsku: HDD, RAM.
1. (3 p.) Jak czyta się „GHz”? Zapisz tę wartość w jednostkach podstawowych SI.
2. (3 p.) Przelicz 1 GiB na kilobajty. (Nie musisz obliczać wyniku, wystarczy podać odp. działanie).
3. (3 p.) Na czym polega grafika rastrowa? (Odpowiedz najwyżej dwoma zdaniami!)
4. (4 p.) Jakie są wady/zalety topologii gwiazdy?
5. (2 p.) Ile jest możliwych adresów IP?
6. (3 p.) Ile pamięci zajmie bitmapa 1000×1000 px zapisana w 256 odcieniach szarości?
7. Zapisujemy kolory w modelu RGB, przeznaczając po 4 b na składową.
 - a) (2 p.) Ile kolorów mamy do dyspozycji?
 - b) (1+1 p.) Jak zapiszemy maksymalną biel, a jak stuprocentową zieleń?
 - c) (6 p.) Opisz cechy koloru, który zostanie zapisany jako: $c_1 = (2, 2, 2)$, $c_2 = (0, 0, 7)$, $c_3 = (15, 2, 0)$.
- *. Dlaczego wszystkie komputery w naszej szkole mają podobne IP? Czemu jest to wygodne?

Kartkówka 4 X

0. (2+2 p.) Rozwiń skrót i napisz, jak to jest po polsku: LAN, I/O.
1. (3 p.) Przelicz 1 MB na kibibity. (Nie musisz obliczać wyniku, wystarczy podać odp. działanie).
2. (3 p.) Jakie są minusy grafiki wektorowej?
3. (2 p.) Uzasadnij nazwę *Internet*.
4. (2 p.) Ile wartości można zapisać na 2 B?
5. (4 p.) Ile zajmuje opis piksela, jeśli zapisujemy obraz:
 - a) czarno-biały, b) w 256 poziomach szarości, c) w modelu o 256 kolorach,
 - d) w modelu RGB, przeznaczając na poziom czerwieni i zieleni po 3 b, a na błękit – 2.
6. (3 p.) Jak w modelu RGB wyglądają zapisy czystych odcieni zieleni?
- *. Ile numerów IP co najmniej marnuje się przy standardowym ich przydzielaniu sieci o 17 urządzeniach?

Kartkówka I – termin II

0. (2 p.) Od jakiego etapu rozwoju ARPANET zaczęto nazywać Internetem? Kiedy to było?
1. (1+2 p.) Wyjaśnij, co to jest MAN i do czego służy DNS.
2. (2 p.) Podaj 5 urządzeń wejściowych.
3. (3 p.) Co jest większe: 1000 MiB czy 8000 Mb?
4. (3 p.) Rozciągnięto obrazek 100×300 px i stał się dwa razy dłuższy. Czy istotnie zmienił się rozmiar pliku, w którym był zapisany, jeśli użyto grafiki: a) rastrowej, b) wektorowej? Uzasadnij!
5. (2 p.) Ile kolorów można zapisać, jeśli do dyspozycji ma się 1 B?
6. (2+2 p.) Ile odcieni czerwieni jest w modelu RGB, w którym na każdy piksel przeznaczono 6 B? Jak wygląda w nim opis koloru białego?
7. (6 p.) Ile pamięci zajmuje opis obrazu 500×1000 px, jeśli zapisano go:
a) jak w zad. 6, b) jako czarno-biały, c) w 256 poziomach szarości?
- 100₍₂₎ *. Dlaczego zwiększanie pojemności pamięci RAM w danym komputerze daje istotne przyspieszenie jego pracy tylko w pewnym zakresie? (Tzn. od pewnej wielkości rozmiar RAM-u nie ma już dużego wpływu na szybkość działania komputera).

PHP – gr. II

Pamiętaj, że chcesz być w stanie odczytać wszystkie kroki Twojego rozumowania lub obliczeń!

1. (7 p.) Oblicz, krótko uzasadniając: $a = 2014 \bmod 2013$, $b = 2013 \bmod 1$, $c = 1 \bmod 2013$,
 $d = 21 \text{ xor } 6$, $e = 21 \text{ xor } 8$, $f = 21 \text{ xor } 17$, $g = 2^{100} \text{ xor } (2^{100} + 5)$.
2. (2,5 p.) Jaką wartość będzie mieć \$n\$ po wykonaniu nast. programu:

```
$n=3;  
for ($i=999; $i>=100; $i=$i-6)  
$n=$n+1;
```
3. (2,5 p.) Podaj ostatnie trzy liczby, które wypisze ten skrypt.

```
for ($i=123; $i<2014; $i=$i+1)  
if ($i%4==0)  
echo $i, "<br />";  
echo $i;
```

* Rozwiąż równanie $101 \bmod x = 11$.

PHP – gr. I

Pamiętaj, że chcesz być w stanie odczytać wszystkie kroki Twojego rozumowania lub obliczeń!

1. (9 p.) Oblicz, krótko uzasadniając: $a = 2013 \bmod 2013$, $q = 2013 \bmod 1$, $b = 1 \bmod 3$,
 $c = 1234567890 \bmod 123456789$, $\acute{c} = 987654321 \bmod 98765432$, $d = 22 \text{ xor } 11$, $e = 22 \text{ xor } 3$,
 $\acute{e} = 22 \text{ xor } 23$, $f = 1234567890987654321 \text{ xor } 1234567890987654321$.

2. (4 p.) Co się tu wypisze?
(Podp.: 20142013 to liczba pierwsza!)

```
$n=0;
for ($i=1; $i<40284026; $i=$i+1)
if (40284026%$i>0)
echo $i;
$n=$n+1;
echo $i;
echo $n;
```

3. (4 p.) Ile wyniosą \$m i \$n po przejściu pętli, jeśli zainicjować je zerami?

```
for ($i=2013; $i>=999; $i=$i-1)
{
if ($i%9==0)
$m=$m+1;
if ($m%9==0)
$n=$n+1;
}
```

MS Excel!

(6 p.) Komórki pustego arkusza Excela wypełniono jak na rys., po czym w komórkę A3 wpisano „=LICZ.JEŻELI(\$A1:A2;"<1")”, a następnie skopiowano jej zawartość do obszaru A3:B4. Co (jakie formuły i jakie wartości) się w nim pojawi?

	A	B
1	0	a
2	1	

(2 p.) Zapisz w M1 formułę, która po skopiowaniu do M1:M123 spowoduje, że w M1 pojawi się liczba trójkąt w obszarze A1:K123, w M2 – trójkąt w obszarze A2:K123, w M3 – trójkąt w obszarze A3:K123 itd.

(4 p.) Ile cyfr ma liczba $1^1 + 3^2 + 5^3 + 7^4 + \dots + 123^n$ (gdzie n jest kolejnym wykładnikiem)?

* Co w zależności od O2 daje formuła

„=JEŻELI(O2>5;JEŻELI(O2<0;3;4);JEŻELI(O2>0;JEŻELI(O2>=2;1);7))”?

MS Excel II

Komórki pustego arkusza Excela wypełniono jak na rys., po czym w komórkę C1 wpisano „=SUMA(A1;\$B2)”, a następnie skopiowano jej zawartość do obszaru C1:D2. Co (jakie formuły i jakie wartości) się w nim pojawi?

	A	B
1	11	
2	1	2

A co da formuła „=JEŻELI(LICZ.JEŻELI(\$A1:B\$2;"<11")>2;1)”? Uzasadnij!

W komórki obszaru D1:Z1234 wpiszę liczby naturalne. Spowoduj, żeby w A1 znalazła się liczba większa o numerach nieparzystych, w których znajduje się jakaś liczba powyżej 44.

I Nowy Rok!...

Gr. 0

0. Ile wartości x się wypisze i jakie będą ostatnie cztery?

```
for (x=101; x<999; x=x+2)
if (x%3>0)
cout << x;
cout << x;
```

1. Ile razy wyświetli się 101?

```
for (i=1; i<2014; i++)
for (j=i/3; j<i; j++)
cout << j << endl;
```

2. Jaka będzie końcowa wartość k i co wyświetli się tutaj?

```
k=0;
for (i=0; i<10; i++)
{
for (j=i; j>-i; j--)
k++;
cout << i << j;
}
```

Gr. 1

0. Ile wyniesie m po przejściu pętli, jeśli wcześniej było zerem?

```
for (i=2014; i>999; i--)
{
if (i%2==0)
if (m%2==0)
m++;
else
m=m+1001;
}
```

1. Ile razy wyświetli się 1?

```
for (i=1; i<100; i++)
for (j=i; j<100; j++)
cout << (j/i==3);
```

2. Jakie powinny być a i b , żeby efekt działania tych pętli był taki sam, ale prawa działała szybciej? Ile mniej więcej razy mniej kroków wykonuje? A dlaczego jest jeszcze szybsza?

```
for (i=0; i<n; i++)
for (j=0; j<n; j++)
if (i<j) k=k+i*i*j;
```

```
for (i=0; i<n; i++)
for (j=a; j>b; j--)
k=k+i*i*j;
```

Poprawa (I?), grupy razem

```
x=0;
y=x;
for (i=1; i<2015; i++)
if (i%2) x=x-i;
else
{
x=x+i;
y=y+1;
}
```

```
for (i=0; i<11; i++)
for (j=9; j>i; j--)
{
cout << j-i;
if (j<4)
cout << endl;
}
```

```
k=0;
for (i=-1000; i<1000; i++)
for (j=-i; j<i; j++)
if (j+i>0)
k++;
```

0. Jakie wartości będą miały zmienne x i y po wykonaniu programu z lewej? Uzasadnij! (Wystarczą dwa krótkie (ale właściwe!) zdania lub odpowiednie działania arytmetyczne).

1. Jak (dokładnie!) będzie wyglądał ekran po zakończeniu działania programu pośrodku?

6*. A jak zmienia się k w programie z prawej?

I... sprawdzian II

W zad. 1 i 2 z przedstawionych rozwiązań powinienem być w stanie odczytać, jak uzyskaliście odpowiedź!

0. (4 p.) Zapisz, nie używając pętli for, fragment programu równoważny następującemu:

```
(for i=2014; i>0; i--) for (j=1; j<99; j++) cout << i; cout << j;
```

1. (5 p.) A co wypisze poniższy programik? (Przypominam, że 201420136 jest liczbą pierwszą!)

```
int i, n=0;
for (i=1; i<40284026; i++)
if (40284026%i>0)
cout << i;
n++;
cout << i;
cout << n;
```

2. (6 p.) Podaj pierwszych sześć i ostatnich sześć liczb, jakie wypisze ten program:

```
int n=0, s=0;
for (int i=1; i<60; i++)
{
    while (s<i)
    {
        n++;
        s=s+n*n;
    }
    cout << s << endl;
}
```

3. (5 p. + możliwe ekstrasy) Napisz program, który wypisze następującą tabliczkę dzielenia całkowitego (jeśli Ci się nie uda, może być (ale o ocenie cel już nie myśl!) mniej równo, bez nagłówek i ew. pierwszej kolumny):

	1	2	3	4	5	6	7	8	9	10
0:	0	0	0	0	0	0	0	0	0	0
1:	1	0	0	0	0	0	0	0	0	0
2:	2	1	0	0	0	0	0	0	0	0
3:	3	1	1	0	0	0	0	0	0	0
4:	4	2	1	1	0	0	0	0	0	0
5:	5	2	1	1	1	0	0	0	0	0
6:	6	3	2	1	1	1	0	0	0	0
7:	7	3	2	1	1	1	1	0	0	0
8:	8	4	2	2	1	1	1	1	0	0
9:	9	4	3	2	1	1	1	1	1	0

4. (5 p.) Napisz programik, który dla podanego $n \geq 10$ przedstawi je jako sumę tylu dziesiątek, ile się w n mieści, oraz ew. reszty. (Czyli np. dla $n=10$ powinno się napisać „10”, dla $n=12$ – „10+2”, dla $n=30$ – „10+10+10”).

1. Co się wyświetli? (Zapisz wszystkie wykonywane rachunki/kroki rozumowania!)

```
int i=0, j=0, k=0, n=0;
for (i=0; i<100; i++)
for (j=i; j<100; j++)
{ while (k<100) { k++; n++; } n++; }
cout << n;
```

2. Dla danego $n \geq 2$ napisz pętlę, która wypisze wszystkie czynniki pierwsze n , każdy tylko raz (czyli dla $n=500$ powinno się wypisać 2 i 5, a dla n będącego dowolną potęgą trójki ≥ 3 – samo 3).

3. Pokaż, jak pracuje (co jak się w niej zmienia i jaki da wynik) nasza funkcja *pierwiastek* dla argumentów (49, 0.2).

4. Napisz programik, który dla podanego naturalnego n napisze na ekranie coś takiego:
a) co dla $n = 5$ wygląda tak:

```
5
54
543
5432
54321
```

b) $1+(2+2)+(3+3+3)+\dots+(k+k+\dots+k)$, gdzie w ostatnim nawiasie jest k składników i k jest największe takie, że cała ta suma nie przekracza n .
(Czyli dla $n = 77$ powinno się napisać „ $1+(2+2)+(3+3+3)+(4+4+4+4)+(5+5+5+5+5)$ ”).

Pamiętaj, że chcę być w stanie odczytać wszystkie kroki Twojego rozumowania lub obliczeń!

```
int f(int n)
{
    // Napisz, jakie warunki pocz. trzeba tu dopisać, żeby działała rekurencja.
    cout << n;
    if (n%3==0)
        return f(n-2)*f(n-2);
    return f(n-1)+1;
}
```

Narysuj drzewo jej wywołań przy obliczaniu $f(8)$. Co się wówczas wyświetli?

Niech $a, b \in \mathbf{N}$. Oblicz $g(a, b)$.

Ile wywołań nastąpi przy obliczaniu $g(2015, 123)$?

```
int g(int a, int b)
{
    if (a*b==0)
        return a;
    return 1+g(a-1, b-1);
}
```

Nie używając gotowych funkcji z bibliotek C++, napisz funkcję, która powie, ile cyfr ma podana jako argument liczba naturalna. Wskazówka: $10 \cdot 10 \cdot 10 \cdot 10 < 12345 < 10 \cdot 10 \cdot 10 \cdot 10 \cdot 10$.

Wskazówka 2: możesz też użyć rekurencji!

* Jeśli masz jeszcze chwilę, możesz napisać rozprawkę nt. „Dlaczego kochamy rekurencję”.

Pamiętaj, że chcę być w stanie odczytać wszystkie kroki Twojego rozumowania lub obliczeń!

```
int f(int n)
{
    cout << n;
    if (n<4) return 0;
    if (f(n-1)%3==1) return f(n-2)*f(n-2);
    return 2*f(n-1)+1;
}
```

Narysuj drzewo wywołań przy obliczaniu $f(6)$. Co się wówczas wyświetli?

Napisz rekurencyjną definicję funkcji g równej sumie cyfr zapisu dziesiętnego jej naturalnego argumentu. (Czyli np. $g(444)=12$, $g(9)=9$, $g(1001)=2$).

Dopisz tutaj(!) warunki początkowe, tak żeby $h(a,b)$ dla całkowitych dodatnich a i b była ich NWD.

Narysuj drzewo wywołań przy obliczaniu $h(120,28)$.

```
int h(int a, int b)
{
    // Tutaj!
    return h(a%b,b%a);
}
```

* Napisz krótki utwór liryczny, w którym oddasz swoje uczucia związane z rekurencją. W cenzuralny sposób!

Grupa 0

0. Zapisz, jak wygląda tablica T po nast. inicjacji: `int T[4][3] = {{1}, {0}, {2, 2, 2}};`

1. Opisz, jak zadziała nast. pętla na n -elementowej tablicy t : `for (int i=n; i>0; i--) t[i-1]=t[n-i];`

2. M jest macierzą $m \times n$.

Napisz fragment programu, który zsumuje wszystkie nieparzyste wartości w jej wierszach o parzystych numerach.

Grupa 1

0. Zapisz, jak wygląda tablica T po nast. inicjacji: `int T[4][3] = {{2, 2, 2}, {0}, {1}};`

... i jak zmieni ją następnie wykonana pętla:

`for (int i=0; i<3; i++) for (int j=0; j<3; j=j+2) T[i][j]=1;`

1. Napisz fragment programu, który wykona transpozycję macierzy $m \times n$, tzn. taką operację, która z

robi $\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 0 & 11 & 12 \end{bmatrix}$.

$$\begin{bmatrix} 1 & 5 & 9 \\ 2 & 6 & 0 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{bmatrix}$$

2. W przypadku jakich macierzy $m \times n$ ciąg poniższych instrukcji będzie miał nieprzewidywalny efekt? Dlaczego? Pokaż, jak zadziałają one na przykładowej dozwolonej macierzy niekwadratowej o 4 wierszach.

`int s=0; for (int i=0; i<m; i++) s=s+M[i][i];`

Wspólne:

6. Zadeklaruj tablicę trójwymiarową, którą można rozumieć jako sześcian zbudowany z 27 sześcianików, i dla każdej jego przekątnej wypisz indeksy sześcianików, przez których środki przebiega.

TABLICE TERMIN[2]

0. Zadeklarowano: `int a, t[6];`

Opisz, co dzieje się w poniższej pętli z a i t , jeśli przed nią do a wpisano: x) 1234, y) 1234567890.

```
for (int i=0; a>0; i++)
{
    t[i]=a%10;
    a=a/10;
}
```

2. Napisz programik, który z danej macierzy $m \times n$:

a) usunie co drugi wiersz i co drugą kolumnę, tj. np. z

$$\begin{bmatrix} 1 & 5 & 9 \\ 2 & 6 & 0 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{bmatrix}$$

powstanie $\begin{bmatrix} 1 & 9 \\ 3 & 11 \end{bmatrix}$.

b) stworzy macierz będącą jej obrazem w symetrii

względem połówiącej ją poziomej prostej, tj. np. z powyższej wyjdzie

$$\begin{bmatrix} 4 & 8 & 12 \\ 3 & 7 & 11 \\ 2 & 6 & 0 \\ 1 & 5 & 9 \end{bmatrix}.$$

1. Jak będzie się zmieniać tablica w tym programie?

```
int t[100]={1,2,3,4,5,6};
for (int i=99; i>0; i--)
    t[i]=t[i]+t[99-i];
```

Niemądry Zbyś próbuje uruchomić coś takiego:

```
int a=10, b, c=1;

void p(int &a, int b)
{
    a = 7;
    b = a;
    cout << 0 << a << b << endl;
}

int f1(int &a)
{
    b = a;
    a++;
    cout << 1 << a << b << endl;
    return c;
}

int f2(int x[2], int y)
{
    int c = y;
    x[1] = y;
    y = x[0];
    cout << 2 << c << x[0] << x[1] << y << endl;
    return c;
}

int main(int argc, char *argv[])
{
    int t[2]={11,22}, a=33, b, c=-2;
    ...
    cout << a << b << c << t[0] << t[1];
    return 0;
}
```

Podaj wszystko, co się wyświetli, jeśli Zbyś w miejsce kropek wpisze:

- a) `p(c, a);`
- b) `p(c, c);`
- c) `p(2, 1);`
- d) `cout << f1(a);`

- e) `cout << f1(b);`
- f) `cout << f1(c);`
- g) `cout << f1(c-1);`
- h) `cout << f1(t[1]);`
- i) `cout << f2(t,c);`
- j) `cout << f2(t,c-1);`

Uwaga: powinienem być w stanie zrozumieć uzasadnienie każdej odpowiedzi! Uwaga': możliwe są odpowiedzi typu: „Nie skompiluje się, bo...”, „przypadkowa wartość, być może program się przerwie, bo...”, „przypadkowa wartość”, „wartość o 1 większa niż to, co wyświetliło się tutaj” (z czytelną adekwatną strzałeczką)!

Niemądra Zosia (na pewno nie z klasy matematycznej!) próbuje uruchomić coś takiego:

```
int t[10]={11,22,33}, a, b=1;

void p0(int &a)
{
    a = -1;
    cout << 0 << a << b;
}

void p1(int x, int &y)
{
    x = 7;
    y = x;
    cout << 1 << a << b << x << y << endl;
}

void p2(int a[10], int b)
{
    a[0]=b;
    b=t[0];
    cout << 2 << t[0] << a[0] << endl;
}

int main(int argc, char *argv[])
{
    int a=-3, b=44, c=2;
    ...
    cout << a << b << c << t[0];
    return 0;
}
```

Podaj wszystko, co się wyświetli, jeśli Zosia w miejsce kropek wpisze:

- a) `p0(a);`
- b) `p0(b);`
- c) `p0(c);`
- d) `p0(t[0]);`
- e) `p1(a,b);`
- f) `cout << p1(a,b);`
- g) `p1(b,b);`
- h) `p1(t[0],t[1]);`
- i) `p2(t,7);`
- j) `p2(t,a);`
- k) `p2(t,b);`
- l) `p2(t,c);`

Uwaga: powinienem być w stanie zrozumieć uzasadnienie każdej odpowiedzi!

Uwaga': możliwe są odpowiedzi typu: „Nie skompiluje się, bo...”, „przypadkowa wartość, być może program się przerwie, bo...”, „przypadkowa wartość”, „wartość o 1 większa niż to, co wyświetliło się tutaj” (z czytelną adekwatną strzałeczką)!

1. Prześledź działanie bisekcji dla sinusa na przedziale (1, 11), jeśli m. zerowe chcemy znaleźć z dokładnością do $\frac{1}{2}$.

2. Jako *calkaR* zapisuję naszą funkcję do całowania metodą pRostokątów, *calkaT* – Trapezów.

Ustal, co jest od czego mniejsze/większe:

$$a = \text{calkaR}(x^2, -1, 0, 0.0001), \quad b = \text{calkaR}(x^2, 0, 1, 0.0001), \\ c = \text{calkaT}(x^2, 0, 1, 0.0001), \quad d = \text{calkaR}(x^2, 0, 1, 0.01), \quad e = \int_{-1}^0 x^2 dx.$$

Uzasadnij! (Może być rysunek + parę słów wyjaśnienia).

3. Jak algorytm stabilny ustali pierwiastki trójmianu $x^2 - 123456x + 0,01$? Dlaczego właśnie tak?

4. Czy przy zapisie zmiennoprzecinkowym błąd: a) względny, b) bezwzględny jest większy w wypadku małych, czy wielkich liczb? Uzasadnij!

5. Zapisz algorytm znajdowania metodą Herona pierwiastka z danej dodatniej liczby w z dokł. do ε .

1. Prześledź działanie Twojej funkcji *bisekcja* dla $f(x) = x$ dla przedziału $(-3, 1)$ i $\varepsilon = \frac{1}{4}$.

2. Czy nasza funkcja obliczająca przybliżenie całki metodą prostokątów da dla funkcji $-x^2$ na przedziale $(0, 1)$ wynik z nadmiarem, czy niedomiarem? Uzasadnij (bez żadnych ogólniejszych faktów), posługując się rysunkiem (ale parę słów objaśnienia się należy)!

3. Bystry Zenek (z klasy matematycznej!) oblicza metodą Herona $\sqrt{10}$, startując od $x = 6$. Jakie będzie x po drugiej zmianie wartości tej zmiennej? (Nie musisz obliczać, wystarczy wyrażenie).

4. Opisz, jak metodą iteracji można (w jakikolwiek sposób) uzyskać dodatni pierwiastek równania $x^{2015} = 7x$. Uzasadnij swe postępowanie.

5. Podaj dwie istotnie różne sytuacje (opisz, jakimi liczbami powinny być a , b i c), w których $a(bc)$ i $(ab)c$ dadzą różne wyniki lub jedno z nich spowoduje przekroczenie zakresu, a drugie nie. Uzasadnij!

INNY SPRAWDZIAN (nie mieszać rozwiązań z poprzednimi, please!)

A. Opisz dwa podejścia zachłanne do problemu plecakowego i podaj po jednym przykładzie (może być dwa razy ten sam) dowodzącym, że nie jest to podejście ogólnie słuszne.

B. Zaprezentuj dynamiczne pakowanie plecaka o pojemności 10, jeśli do wyboru mamy (rozpatrywane w tej kolejności) przedmioty: $p_1(5,10)$, $p_2(4,40)$, $p_3(6,30)$, $p_4(3,50)$. (Podaję (objętość, wartość)). Utwórz również tablicę do odczytania optymalnego zestawu i pokaż, jak go odczytać.

C. Podaj przykład danych w problemie wydawania reszty, przy których algorytm zachłanny:

C1. „zatnie się”, tzn. nie da rozwiązania, mimo że takie istnieje;

C2. da odpowiedź (wyda resztę), ale nieoptymalnie.

0. Jak będzie wyglądała tablica (99, 98, 97, ..., 2, 1, 100) po 20., a jak po 60. kroku algorytmu sortowania:

a) bąbelkowego, b) przez proste wstawianie, c) przez wstawianie połówkowe, d) przez proste wybieranie?

1. Opisz wszystkie możliwe ustawienia liczb 1, 2, 3, ..., 100 w ciąg, żeby w algorytmie min+max wykonało się możliwie najmniej instrukcji podstawienia. Uzasadnij!

2. Znajdź liczbę porównań i liczbę podstawień wykonywanych przy sortowaniu przez proste wstawianie i proste wybieranie posortowanego ciągu.

6. Pokaż wygląd tablicy (1, 8, 5, 2, 3, 0, 2, 3, 7, 6) po każdym „dużym kroku” najbardziej zaawansowanego sortowania bąbelkowo (tzn. biegamy dwukierunkowo i pamiętamy, których elementów nie trzeba już porównywać).

1. (5 p.) Czy QS jest algorytmem stabilnym? Uzasadnij!

2. (5 p.) Prześledź quicksorta (co jest pivotem, co się wywołuje, co się gdzie wpisuje, ...) dla tablicy (1, 2, 3, 4, 5, 0).

3. (10 p.) Pokaż kolejne wywołania zstępującego mergesorta dla tablicy (0, 1, 2, ..., 9). Jakie są kolejne scalenia?

6. Jaka jest złożoność Qsorta dla już posortowanej tablicy? Podaj kolejno wykonywane operacje i je zlicz.

Z0. (3 p.) Co wyświetli poniższy fragment programu, jeśli zdradzę, że 78 to kod znaku 'N'? Krótko uzasadnij, nie używając więcej danych, niż potrzeba. `char z=78; cout << int('M') << char(z-2) << char(z-'2') << int(z-'P') << char(z-'L') << char('x'-'M'+z);`

Z1. (6 p.) Napisz funkcję, która dla podanych dwóch napisów zwróci ich *longest common prefix*, przy czym prefiksem danego łańcucha nazywamy jego dowolny początkowy podłańcuch (czyli prefiksami napisu „ABC” są: „” (napis pusty), „A”, „AB” i „ABC”).

Z2. (6 p.) Co robi poniższa funkcja? Możesz ustalić jej wartości dla łańcuchów ABC, ABC01, ABC0032, 456, 77ABC, 88ABC i 00ABC, ale wniosek/odpowiedź powinny być ogólne.

```
int f(string s) {
    int k=0;
    for (int i=0; i<s.size(); i++) if (s[i]>'0' && s[i]<='9') k=k+s[i]-'0';
    return k; }
```

Z'. A gdyby w warunku dopisać jeszcze „&& (s[i]-'0')%2==1”?

Z3 dla gr. 1 (6 p.) Dana jest wypełniona Twoim algorytmem LCS tablica *T* długości częściowych LCS-ów napisów *n1* i *n2* (z uwzględnieniem prefiksów pustych lub nie – jak Ci wygodniej). Zapisz część programu odpowiadającą za znalezienie ich (dowolnego) LCS-a.

Z3.2 (6 p.) Co powinien wpisać nasz algorytm LCS w pustą komórkę tablicy, jeśli odpowiada ona porównaniu liter a) A i A, b) A i B?

3.1

∅	∅
A	

3.2

∅	B
A	

3.3

A	BA
A	

3.4

AA	AA
AAB	

Z0. Co wyświetli poniższy fragment programu? Krótko uzasadnij, nie używając więcej danych, niż potrzeba.

```
char z='t'; cout << int(z) << char(z+1) << int(z+'K'-'m') << char('K'-'T'+z);
```

Z1. Napisz funkcję, która dla danego łańcucha złożonego z samych cyfr zwróci „ich sumę”. (Tzn. dla napisu „1021” da wartość 4).

Z2. Napisz funkcję (czytelną, złożoność nie gra roli), która dla podanych dwóch napisów stwierdzi, czy:

a) pierwszy jest podłańcuchem drugiego (można założyć, że pierwszy jest krótszy niż drugi);

b) są anagramami.

* Podaj pesymistyczne złożoności obu tych funkcji (z uzasadnieniem!).

∅	B
A	?

Z3. Czy możliwe, żeby w tabelce LCS wystąpił taki fragment jak obok, jeśli komórka oznaczona „?” odpowiada porównaniu liter A i B?

0. (za 0 pkt, ale niezrobienie go oznacza ndst z zad. dom.!)

Zapisz procedurkę „odwrotności” z zad. dom. (która znajduje odwrotności danego elementu Z_n , ew...).

1. Podaj graf, w którym w zależności od punktu startowego algorytm zachłanny może znaleźć komiwojażerowi różne cykle. (Uwaga: ABCDA, BCDAB i ADCBA to oczywiście ten sam cykl!)

2. Na Wyspach Bergamutach w obiegu są monety 1, 2, 5, 9 i 15 ytołzów. Pokaż, jak algorytm zachłanny wyda kwotę 19 ¥. Udowodnij naszym właściwym algorytmem, że jest to wydanie optymalne. Czy w bergamuckim systemie monetarnym najlepsze wydanie zawsze można znaleźć zachłannie?

Podaj [inny] przykład takiego systemu.

3. Wypełniamy tabelkę, żeby stwierdzić, jaką maksymalnie wartość może mieć zawartość pakowanego plecaka. Jesteśmy akurat w kolumnie 12 l i wierszu odpowiadającym megapiłkom do rugby o wartości 99 zł i objętości 3 l. Zakładamy, że nie jest to pierwszy wiersz i wszystko, co trzeba, zostało już odpowiednio wypełnione. Co należy sprawdzić i jak zareagować, żeby wpisać do tablicy odpowiednią wartość?

* Zapisz (ściśle, nie literacko!) algorytm rozwiązujący ciągły problem plecakowy (okazuje się, że wtenczas zachowanie zachłanne jest właściwe) przy danych: n – liczba substancji do ukradzenia, ich dostępne ilości V_i , jednostkowe wartości substancji w_i , V – pojemność naszego tankowca.

MATURA 2015/I

Z1. Czy strategia D' – „Wybierz film, który kończy się najwcześniej, a jeśli jest takich więcej, to wybierz spośród nich losowo.” – jest w problemie telewizora optymalna? Krótko uzasadnij!
Czy suma długości obejrzanych filmów jest w tej strategii zdeterminowana? Krótko uzasadnij!

Z22. Jaki rodzaj kompresji jest stosowany w plikach BMP?

Z3. Co jest efektem działania rozszerzonego alg. Euklidesa? Podaj jego przykładowy wynik dla pary (10, 2).

1. Oblicz naszym algorytmem pole wielokąta dla $P_0, P_1, \dots = (1, 0), (2, 2), (0, 2), (1, 1), (0, 1)$.

2. Zapisz niegłupi algorytm stwierdzający, ile punktów wsp. mają proste AB i OP , gdzie $O = (0, 0)$, a wsp. pktów A, B i P są danymi liczbami całkow. $x_A, y_A, \dots$

3. Z klawiatury podam kolejno: wsp. (różnych) punktów krat. P i Q , liczbę nat. n i n par współrz. różnych punktów krat. Napisz program, który stwierdzi, czy po obu stronach prostej PQ leży tyle samo podanych punktów krat.

6. Dane są współrzędne wierzchołków n -kąta wyp., ale w nieznanej kolejności. Jak możliwie mądrze (oszacuj pesymistyczną złożoność) ustawić je w taki ciąg, który wyznacza brzeg wypukłego n -kąta? (Założmy tym razem, że operacje zmiennoprzecinkowe wykonują się dokładnie).

1. (6 p.) Podaj (z uzasadnieniem) w $\mathbf{Z}_8$: $a = -3$, $b = -0$, $c = -4$, $d = 3^{-1}$, $e = 0^{-1}$, $f = 1^{-1}$.

2. (2 p.) Czy w $\mathbf{Z}_{2015}$ element 44 jest odwracalny?

3. (4 p.) Rozwiąż w $\mathbf{Z}_{200}$ równania: $x \cdot 150 = 50$, $y \cdot 15 = 11$.

4. (4 p.) Czy z tekstu BACADABUCA można otrzymać CABAUACDBA, jeśli szyfrujemy:
a) podstawieniowo (monoalfabetycznie)? b) przestawieniowo?

5. (3 p.) Ile jest szyfrów podstawieniowych monoalfabetycznych? Dlaczego nie wszystkie są zbyt sensowne?

MATURA 2010/I

Z1. (2 p.) Algorytmem opisanym w zadaniu szyfrujemy wiadomość długości 1234. Który jej znak będzie sześćdziesiątym siódmym znakiem szyfrogramu?

Z2. (2 p.) Zaprezentuj działanie algorytmu „tablica zero-jedynkowa” dla tablicy o 192 zerach.

Z3b. (2 p.) Jaka jest pesymistyczna złożoność pętli: dopóki $a > 0$ { $a \leftarrow a \bmod b$; $b \leftarrow b - a$ } ?

Jeśli to za trudne, podaj jej działanie dla danych, przy których Twoim zdaniem będzie kręcić się strasznie długo.

Z3e. (1 p.) Na czym polega grafika rastrowa?

1. Czy MergeSort w wersji zstępującej jest stabilny? Uzasadnij!
2. Dana jest procedura $Merge(a, b)$ scalająca fragment tablicy od pozycji a do pozycji b , jeśli jego „połówki” są posortowane.
- x) Zapisz MergeSorta.
- y) Jakie jej wywołania wykona kolejno wstępujący MergeSort na tablicy 200-elementowej?
3. Sortujemy naszym QuickSortem tablicę $(0, 1, 2, \dots, 9)$.
- a) Narysuj drzewko wywołań. (Przyjmijmy, że początkowe oznaczamy jako $Q(0,9)$).
- b) Ile podstawień wykonamy? Uzasadnij!

1. (8 p.) Zapisz w C++ funkcję, która dla podanej liczby nat. $b > 1$ i zapisu liczby w systemie o podstawie b poda wartość tej liczby (czyli np. $f(8, "70") = 56$). Uwzględnij systemy o bazach > 10 i cyframi 10, 11, ... niech będą wówczas – jak Ci wygodnie – małe lub duże litery. (Bez tego – maks. 4 pkt).

2. (4 p.) Jeśli $l = 12345012345_6$, to jak wygląda szóstkowy zapis liczb: $x = l \bmod 36$, $y = l \div 6^5$?

3. (5 p.) Jak wygląda zapis little endian liczby $256^2 + 9$ na 4 B?

4. (10 p.) Z pomocą Euklidesa znajdź $17^{-1} \bmod 100$. Jeśli nie umiesz, oblicz chociaż (za maks. 4 pkt) $\text{NWD}(18, 102)$.

5. (6 p.) Jakie mnożenia wykonają się przy szybkim podniesieniu liczby a do potęgi 99?

6. (5 p.) Dlaczego w RSA $e = 1$ nie jest stosowane?
Jakie jest najmniejsze możliwe e dla $p = 7$, $q = 13$? Uzasadnij!

7. (8 p.) Kluczem publicznym w RSA jest $(5, 85)$. Zaszzyfruj liczbę 70.

8. (2 p.) Co to jest współczynnik kompresji?

9. (10 p.) Jaki tekst złożony z liter A, B, C po kompresji RLE ma postać 7ACBC2A?
Zapisz w C++ algorytm, który dokona takiej kompresji dowolnego łańcucha, wypisując rezultat na ekran.

0*. Zapisz w C++ algorytm dekompresji wg algorytmu z zad. 9 przy założeniu, że kompresowane teksty składają się tylko z liter A, B, C i grupowane są one powyżej po 9.

Uwagi: 1-2. jeśli nie radzisz sobie ze stringami, możesz napisać procedurki, które piszą coś na ekranie (ale za mniej pktów); w zad. 4 nie musisz używać C++.

0. (2 p.) Do jakich typów szyfrów można zaliczyć szyfr Vigenère'a? Uzasadnij!

1. (4 p.) Szyfrujemy tekst, przepisując co trzeci jego znak począwszy od pierwszego, potem co trzeci począwszy od drugiego, i co trzeci począwszy od trzeciego (czyli dla tekstu „1234567” uzyskalibyśmy „1472536”). Napisz w C++ funkcję, która zdekoduje uzyskany tak szyfrogram.

2. Kompresujemy sposobem RLE tak, że w szyfrogramie jest na zmianę cyfra i znak wiadomości (czyli może to np. być „9A9A3A1B1A43”).

a) (1 p.) Oblicz wsp. kompresji tekstu zakodowanego w przykładzie.

b) (6 p.) Zapisz takie kodowanie jako funkcję w C++. (Jej wartością ma być oczywiście szyfrogram).

3. (4 p.) Zapisz w C++ procedurę, która dla argumentów $l \in \mathbb{N}_+$ i $b \in \{2, 3, \dots, 35\}$ wypisze na ekran zapis liczby l w systemie o bazie b , znajdując go z definicji i cyfry pow. 9 zapisując jako odpowiednie litery.

4. (4 p.) Zapisz algorytm szybkiego potęgowania nieużywający pomocniczej tablicy.

5. (4 p.) Czy istnieją całkowite α i β , dla których: a) $\alpha \cdot 6^{123} + \beta \cdot 35 = 25$, b) $\alpha \cdot 6^{123} + \beta \cdot 45 = 3$? Uzasadnij!

6. (2 p.) W algorytmie RSA kluczem jawnym jest $(3, 55)$. Znajdź klucz tajny. (Nie musisz stosować algorytmu, ale uzasadnij odpowiedź).

1. Sformułuj precyzyjnie (podając specyfikację) dowolny wariant problemu komiwojażera i uzasadnij, że postępowanie zachłanne go nie rozwiązuje.

2. Zapisz zachłanny algorytm wydawania reszty r dla zestawu nominałów $n_1 < n_2 < \dots < n_k$.

Podaj przykład takiego systemu:

- a) z możliwie najmniejszym k , gdzie algorytm zachłanny nie jest optymalny, ale zawsze da rozwiązanie,
- b) gdzie $k = 7$, a algorytm zachłanny dla dowolnej r daje rozwiązanie optymalne.

3. Zaprezentuj dynamiczne pakowanie plecaka o pojemności 5, jeśli do wyboru mamy (rozpatrywane w tej kolejności) przedmioty (podaję (objętość, wartość)): $p_1(2, 3)$, $p_2(3, 4)$, $p_3(4, 5)$, $p_4(5, 6)$, każdy w jednym egzemplarzu!

1. Wiedząc, że wyznacznik wektorów to znakowane pole odpowiedniego trójkąta, udowodnij poprawność naszego algorytmu obliczania pola czworokąta.

2. Dane są współrzędne (x_1, y_1) , (x_2, y_2) , ..., (x_n, y_n) kolejnych wierzchołków n -kąta wypukłego oraz punkt (x_0, y_0) . Zapisz algorytm stwierdzający, czy leży on wewnątrz tego n -kąta. (Uwaga: powyżej oceny db+ ma znaczenie jego złożoność!)

3. W 2015. levelu wersji β najnowszej gry komputerowej firmy \$liv Games przez planszę biegnie mur będący prostą o równaniu $Ax + By = C$, gdzie A , B i C są całkowite, a w punkcie kratowym (x_0, y_0) siedzi bezlitosny Snajper (Nauczyciel z poprzedniego levelu). Zapisz algorytm pozwalający stwierdzić, czy punkt kratowy (x_1, y_1) jest w jego zasięgu. (A z każdego punktu muru widać wszystko).

6. Dane są współrzędne wierzchołków n -kąta wyp., ale w nieznanym porządku. Jak możliwie mądrze (oszacuj pesymistyczną złożoność) ustawić je w taki ciąg, który wyznacza brzeg wypukłego n -kąta? (Założmy tym razem, że operacje zmiennoprzecinkowe wykonują się dokładnie).

MATURA 2015/I.2

1. (4 p.) Co stoi w miejscach (1)-(6) w zapisie algorytmu z zad. o filmach?

Specyfikacja:

Dane:

T – zbiór filmów z programu telewizyjnego z godzinami emisji i czasami ich trwania,

S – strategia z tabeli 1.

Wynik:

P – (1).

Algorytm:

Krok 1. Zainicjuj P jako (2).

Krok 2. Dopóki T zawiera jakieś filmy, wykonuj:

. stosując strategię S , wybierz z (3) film x i usuń go z T

. dodaj film x do zbioru P

. usuń z (4) wszystkie filmy, których czasy emisji kolidują z czasem emisji (5).

Krok 3. Zakończ wykonywanie algorytmu i wypisz wszystkie filmy z (6).

2. (2 p.) Co wiesz o formacie BMP?

3. (4 p.) Zapisz listę kroków rekurencyjny algorytm Euklidesa obliczania NWD całkowitych dodatnich liczb x i y . (Nie musi być tak jak w zadaniu!)

MATURA 2010/I.2

1. (4 p.) Przy oznaczeniach jak w zadaniu (tj. d – długość tekstu do zaszyfrowania, $tekst[1 \dots d]$ – tablica zawierająca tekst do zaszyfrowania, $szyfr[1 \dots s]$ – tablica zawierająca tekst po zaszyfrowaniu) zapisz algorytm ustalający, na jakiej pozycji w tablicy $szyfr$ znajdzie się $tekst[n]$.
2. (4 p.) Zapisz algorytm (niekoniecznie taki jak w zadaniu) rozwiązujący zadanie „Tablica zero-jedynkowa” (tzn. znajdujący liczbę zer w tablicy $a[1 \dots 1023]$, jeśli wszystkie zera...) metodą połowienia.
- 3a. (1 p.) Co to jest stos?
- 3b. (3 p.) Zapisz nie schematem blokowym algorytm obliczający NWD metodą jak w zadaniu (tzn. iteracyjnie, jak Euklides, odejmowaniem).

MATURA 2015/II

- 4a. (4 p.) Napisz funkcję, która da resztę z dzielenia przez 4 liczby naturalnej, jeśli argumentem jest jej zapis binarny.
- 4b. (6 p.) Dla danych dwóch zapisów binarnych liczb naturalnych długości 22 stwierdź, który przedstawia większą liczbę. (Zapis żadnej liczby dod. nie zaczyna się zerem).

Uwaga: jak zwykle oczekuję zapisu wszelkich rozumowań i rachunków, jakie wykonujesz!

1. (4 + 5 p.) Zapisz $7/_{27}$: a) trójkowo, b) szóstkowo.
 2. (3 + 5 p.) Zapisz czwórkowo: $x = C0B1234D_{16}$, $y = 21/2^{201}$.
 3. (2 p.) Ile wartości można zapisać na 12 b?
 4. (4 p.) Co to jest G_i , a co G ? (Podaj ew. wieloznaczności).
 5. (9 p.) Jaki jest zakres kodowania: a) ZM na 2 B, b) U1 na 1 B, c) U2 na tetradzie?
 6. (12 p.) Zapisz w ZM, U1 i U2 na 4 B: -3 , 11 , -2^{11} .
 7. (6 p.) Co oznacza zapis 1100.1001 w kodzie ZM/U1/U2?
- 0*. Zapisz w C++ funkcję o argumentach $l1$ i $l2$ będących 40-bitowymi zapisami ujemnych liczb całkowitych w U1, która zwraca 40-bitowy zapis ich sumy w U1. (Można założyć, że się zmieści).

Z1. (5 p.) Jaka jest najmniejsza możliwa wartość M , która spowoduje, że pseudokod w ramce będzie definicją funkcji o argumentach naturalnych?

Uzasadnij!

Wyznacz $f(4)$, $f(5)$, $f(6)$, $f(7)$, $f(8)$, $f(9)$.

Podaj wzór na liczbę wykonań elementarnych przy wywołaniu $f(n)$.

funkcja $f(i)$

jeżeli $i < M$

zwróć 2 i zakończ;

w przeciwnym razie

jeżeli $i \bmod 3 > 0$

zwróć $f(i - 2) + f(i - 1)$

w przeciwnym razie

zwróć $f(i - 3) - 1$

Z2. (2 p.) Według algorytmu z zad. 2 znajdź 100 cyfr po przecinku rozwinięcia trójkowego liczby $2/_{99}$.

Z64. (5 p.) Ile co najmniej bitów zostało przekłamanych, jeśli obrazek jest zapisany o tak:

Uzasadnij! Podaj, które to mogą być bity (wszystkie możliwości!).

1101010
1110011
0100111
0001010
1101011
1001111

Z65.1. (5 p.) Co wyświetli taka linijka C++?

```
cout << 7/3 << 7*1.0/3 << float(7)/3 << float(7/3) << 7/3*1.0;
```

M65.2. (5 p.) Zapisz w C++ funkcję stwierdzającą dla całkowitych a , b , c , d , czy a/b to mniej niż c/d .

I1. (4 p.) Zapisz algorytm, który dla danego posortowanego niemalejąco ciągu liczbowego $a_1, a_2, \dots, a_n$ i wartości x poda $\max\{k \in \{1, 2, \dots, n\} : a_k < x\}$, jeśli takie k istnieje, a 0, jeśli go nie ma, wyszukując wynik połówkowo.

I2. (4 p.) Podaj postać ciągu po kolejnych obrotach zewn. pętli sortowania przez wstawianie ciągu (8, 4, 6, 9, 1, 3, 3). Na co wpływ ma, czy sortujemy z wyszukiwaniem liniowym, czy binarnym?

I3. (2 p.) Która liczba jest większa: $1010.1111.0000.0000.1111.0101.0001.0011.1100.1010_2$ czy 12740057211712_8 ? Uzasadnij!

II0. (3 p.) T jest 12345-elementową tablicą zmiennych typu int. Wypisz (w dow. kolejności) wszystkie jej elementy występujące więcej niż raz (każdy tylko raz). Złożoność czasowa dowolna, ale z pamięcią proszę realistycznie – nie iść w gigabajty!

II. Zapisz w C++:

1. (3 p.) funkcję, która dla zmiennej typu string zwróci długość najdłuższego jej podłańcucha złożonego z cyfr; (np. $f(„terefere”) = 0$, $f(„123”) = f(„12abc101p”) = f(„k999o999o21”) = 3$).

2) (3 + 2 p.) procedurę, która dla napisu złożonego z małych liter alfabetu wypisze na ekran w porządku alfabetycznym wszystkie jego litery (każdą tylko raz) z częstością ich wystąpień (np. dla napisu „kasa” powinno pojawić się coś w rodzaju „a2k1s1”, mniejsza o czytelność); uwaga: na ocenę cel algorytm musi mieć możliwie małą złożoność!

3. (3 p.) funkcję *dzielisie*, która ustala, czy suma cyfr z pozycji o nrach parzystych (czyli zawieramy obie skrajne) danego jako argument PESEL-u (typ dobierz wg uznania) dzieli się przez 17.

0. (4 p.) Które z podanych na tablicy grafów są izomorficzne? Uzasadnij izomorfizm[y] oraz ich brak!

1. (2 + 3 + 4 p.) Narysuj:

a) ukorzenione drzewo binarne o 5 liściach (= wierzchołkach, które nie mają potomków) i minimalnej liczbie wierzchołków,

b) wszystkie lasy o 4 wierzchołkach, każdy tylko raz!

c) drzewo wyrażenia: $x) p \Rightarrow \neg((q \wedge r) \vee (s \Rightarrow \neg t))$, $y) + -ab - *c + def$.

2.gr.1: (6 p.) Jaka jest złożoność tarnowskiego algorytmu sortowania kubelkowego? Uzasadnij!

2.gr.2: (4 p.) Zapisz funkcję o złożoności pierwiastkowej sprawdzającą pierwszość jej argumentu.

67. (3 p.) Napisz procedurę, która dla podanego jako argument napisu długości $\leq n$ „wyrówna go do prawej”, dopisując z przodu tyle zer, żeby jego długość wyniosła n . (Czyli przy $n = 5$ napis „011” powinien się zmienić na „00011”, napis „0” – na „00000”, a napis „12345” – pozostać niezmienny).

(3 p.) W zad. „Bruker” zmień system pracy firmy na nast.: „wszyscy pracownicy (21 osób) układają kostkę granitową do dnia, w którym rano zapas kostki granitowej jest mniejszy niż ten zużywany codziennie przez cały zespół 21 pracowników. Wówczas 13 pracowników zostaje oddelegowanych do pracy z kostką bazaltową, a reszta układa nadal kostkę granitową. Kiedy zapas kostki granitowej zostanie uzupełniony (rano zapas kostki wystarczy na cały dzień pracy dla wszystkich 21 pracowników), wszyscy ponownie układają tylko kostkę granitową.” i wypisz do pliku dwie kolumny z podpktu d, czyli „obrazujące poranny zapas kostki granitowej i kostki bazaltowej dla danych z okresu od 1.03.2013 do 1.11.2013 włącznie”.

Z3. Ile mnożeń wykona się w algorytmie „ciekawego mnożenia” (przypominam: jeżeli $n = 1$, podaj wynik x i zakończ, w przeciwnym razie jeżeli $n \bmod 3 = 0$, $\{k \leftarrow F(x, n \div 3)$, podaj wynik $k*k*k$ i zakończ}, w przeciwnym razie podaj wynik $x*F(x, n-1)$ i zakończ) dla $n = 6, 11, 100, 3^m+2$?

Z4. Zapisz silniowo 2016.

Jaka jest najmniejsza liczba, która silniowo wymaga 5 cyfr?

Jaka jest największa cyfra, która może stać na pozycji n od prawej zapisu silniowego, jeśli numerujemy od zera?

Z52. Ile kolorów daje się zapisać w modelu RGB, w którym plik 200×300 px zajmuje blisko 100 kB?

Z69. Napisz funkcję, która dla danego jako jej argument napisu t zwróci długość najdłuższego jego podłańcucha, który zaczyna się od F i składa z kolejnych wielkich liter alfabetu.

(Dla $t = „FGDFHKFFGGHHKFGF”$ chodziłoby o któryś z trzech łańcuchów „FG”).

103. Napisz program, które wypisze w kolejności alfabetycznej z bazy nazwiska wszystkich pacjentów (każdego tylko raz), którzy mieli kiedykolwiek „RTG zeber”. (Uwaga: program powinien działać również przy innych danych!)

Podaj 10-znakowy genotyp odporny (taki, który czytany od strony lewej do prawej oraz od strony prawej do lewej ma dokładnie taką samą część kodującą (czyli złożoną z podłańcuchów zaczynających się od „AA” i kończących „BB”)).

1. Czy jeśli ciąg stopni wszystkich wierzchołków grafu G jest po uporządkowaniu taki sam jak uporządkowany ciąg stopni wszystkich wierzchołków grafu H , to są one izomorficzne? Uzasadnij!
2. Narysuj wszystkie ukorzenione drzewa o 4 wierzchołkach (każde tylko raz).
3. Ile wierzchołków może mieć las o 11 krawędziach? Krótko uzasadnij.
4. Narysuj drzewo wyrażenia $\#*ab*\#cdef\#\#fgh$, gdzie $\#$ oznacza pewne działanie binarne, a $*$ - trójargumentowe.
5. Zapisz w notacji polskiej wyrażenie o danym (na tablicy) drzewku.
- 6.1. Podaj dwa skrajne przypadki tarnowskiego sortowania kubełkowego i opisz, jak wówczas działa.
- 6.2. Zapisz funkcję o złożoności pierwiastkowej sprawdzającą doskonałość jej argumentu.