

Programowanie zaawansowane FM i IN 2017/2018

Zadania domowe – seria 1.

Termin: 8 maja 2018 r.

Autorem wszystkich zadań jest dr Przemysław Olbratowski.

Rozwiązaniem każdego z zadań powinien być plik lub więcej plików zawierających kod w języku C++, mających standardowe rozszerzenia, np. `.cpp`, `.h`. Nazwa pliku powinna mieć postać `nazwisko-zadanie.roz`, np. `kowalski-calc.cpp`. W przypadku istnienia kilku plików o tych samych rozszerzeniach należy jeszcze dodać do nazwy ich numery: `nazwisko-zadanie-numer.roz`, np. `kowalski-calc-1.cpp`, `kowalski-calc-2.cpp`.

Zadanie 1. `calc` – Kalkulator pięciodziałaniowy (0,5 pkt.)

Napisz program `calc` implementujący kalkulator, który wykonuje cztery podstawowe działania arytmetyczne oraz wyciąga pierwiastek kwadratowy. Po uruchomieniu program wczytuje ze standardowego wejścia liczbę 1, 2, 3, 4, lub 5, definiującą działanie, które ma zostać wykonane: odpowiednio dodawanie, odejmowanie, mnożenie, dzielenie i pierwiastkowanie. Następnie program wczytuje argumenty wybranego działania w odpowiedniej ilości i wypisuje na standardowe wyjście jego wynik. Jeżeli którakolwiek daną podano niepoprawnie, program wypisuje komunikat o błędzie. Użyj instrukcji `switch`.

Przykładowe wykonanie

Wejście:

```
2
7.5 5.2
```

Wyjście:

```
2.3
```

Zadanie 2. `barometric` – Wzór barometryczny (0,5 pkt.)

Wzór barometryczny

$$h = -\frac{RT}{g\mu} \log\left(\frac{p}{p_0}\right),$$

gdzie $R = 8,3144598$, $g = 9,80665$, $\mu = 0,0289644$, zaś $p_0 = 1013,25$, wiąże wyrażoną w metrach wysokość h nad poziomem morza z ciśnieniem atmosferycznym p , wyrażonym w hektopaskalach, oraz temperaturą powietrza T , wyrażoną w kelwinach. Napisz funkcję `barometric`, która przyjmuje p i T oraz zwraca h . Korzystając z tej funkcji, napisz program `barometric`, który wczytuje ze standardowego wejścia p i T oraz wypisuje na standardowe wyjście obliczoną wartość h .

Przykładowe wykonanie

Wejście:

```
955 290
```

Wyjście:

```
502.596
```

Zadanie 3. leap – Lata przestępne (1 pkt.)

Według kalendarza gregoriańskiego przestępne są lata podzielne przez 4 z wyjątkiem lat podzielnych przez 100, ale niepodzielnych przez 400. Napisz program `leap`, który wczytuje ze standardowego wejścia rok i wypisuje na standardowe wyjście `Rok (r) jest przestępny.`, jeśli jest on przestępny, lub `Rok (r) nie jest przestępny.` w przeciwnym przypadku (w obu przypadkach w miejsce `(r)` należy wstawić rok).

Przykładowe wykonanie

Wejście:

2000

Wyjście:

Rok 2000 jest przestępny.

Zadanie 4. rnd – Zaokrąglanie liczb (0,5 pkt.)

Zdefiniowana w pliku nagłówkowym `cmath` funkcja `round` zaokrągla podaną liczbę rzeczywistą do najbliższej liczby całkowitej; na przykład `round(3.14159)` daje w wyniku 3. Korzystając z tej funkcji napisz funkcję `rnd`, która przyjmuje liczbę rzeczywistą oraz liczbę cyfr po przecinku i zwraca podaną liczbę rzeczywistą zaokrągloną do podanej liczby cyfr po przecinku. Wykorzystując napisaną funkcję napisz program `rnd`, który wczytuje ze standardowego wejścia liczbę rzeczywistą oraz liczbę cyfr po przecinku i wypisuje na standardowe wyjście podaną liczbę rzeczywistą zaokrągloną do podanej liczby cyfr po przecinku.

Przykładowe wykonanie

Wejście:

3.14159 3

Wyjście:

3.142

Zadanie 5. fraction – Część całkowita i ułamkowa (0,5 pkt.)

Napisz funkcję `fraction`, która przyjmuje jako argumenty zmienną rzeczywistą oraz referencje dwóch zmiennych rzeczywistych i wpisuje do tych zmiennych odpowiednio część całkowitą (cechę) oraz ułamkową (mantysę) wartości otrzymanej jako pierwszy argument. Korzystając z tej funkcji napisz program `fraction`, który wczytuje ze standardowego wejścia liczbę rzeczywistą i wypisuje na standardowe wyjście jej część całkowitą oraz ułamkową.

Przykładowe wykonanie

Wejście:

3.141592

Wyjście:

3 0.141592

Zadanie 6. dms – Stopnie, minuty i sekundy (1 pkt.)

W geografii kąty wyraża się zwykle podając całkowitą liczbę stopni, minut i sekund. Napisz funkcję `dms`, która przyjmuje jako argumenty zmienną zawierającą niecałkowitą liczbę stopni oraz referencje trzech zmiennych całkowitych i wpisuje do tych zmiennych całkowite liczby stopni, minut oraz sekund. Korzystając z tej funkcji napisz program `dms`, który wczytuje ze standardowego wejścia niecałkowitą liczbę stopni i wypisuje na standardowe wyjście odpowiednio całkowite liczby stopni, minut i sekund.

Przykładowe wykonanie

Wejście:

123.37

Wyjście:

123 22 12

Zadanie 7. quadratic – Równanie kwadratowe (0,5 pkt.)

Napisz funkcję `quadratic` rozwiązującą równanie kwadratowe $ax^2 + bx + c = 0$. Funkcja przyjmuje trzy zmienne rzeczywiste, reprezentujące wartości parametrów a , b , c , oraz referencje dwóch zmiennych rzeczywistych i zwraca wyróżnik równania. Jeżeli równanie posiada rozwiązania, funkcja wpisuje je do otrzymanych referencji (jeśli istnieje tylko jedno rozwiązanie, jest wpisywane do pierwszej z referencji). Korzystając z tej funkcji napisz program `quadratic`, który wczytuje ze standardowego wejścia parametry a , b i c oraz wypisuje na standardowe wyjście wyróżnik równania. Ponadto, jeśli równanie posiada rozwiązania, program wypisuje je na standardowe wyjście.

Przykładowe wykonanie

Wejście:

2 -7 -15

Wyjście:

169

-1.5 5

Zadanie 8. swap – Zamiana wartości zmiennych (0,5 pkt.)

Napisz funkcję `swap`, która przyjmuje jako argumenty referencje dwóch zmiennych rzeczywistych i zamienia ich wartości. Funkcja powinna być przystosowana do użycia w następującym programie:

```
#include <iostream>
using namespace std;

int main()
{
    double a = 2.71828, b = 3.14159;
    swap(a, b);
    cout << a << " " << b << endl;
    return 0;
}
```

Wykonanie

Wyjście:

3.14159 2.71828

Zadanie 9. less – Mniejsza z dwóch liczb (0,5 pkt.)

Napisz funkcję `less`, która przyjmuje jako argumenty referencje dwóch zmiennych rzeczywistych i zwraca referencję tej z nich, która ma mniejszą wartość. Funkcja powinna być przystosowana do użycia w następującym programie:

```
#include <iostream>
using namespace std;

int main()
{
    double a = 3.1, b = 7.2;
    less(a, b) = 10.3;
}
```

```

    cout << a << " " << b << endl;
    return 0;
}

```

Wykonanie

Wyjście:

10.3 7.2

Zadanie 10. triangle – Trójkąt z gwiazdek (0,5 pkt.)

Napisz program `triangle`, który wczytuje ze standardowego wejścia liczbę naturalną n i drukuje na standardowe wyjście trójkąt z gwiazdek mający n wierszy i n kolumn.

Przykładowe wykonanie

Wejście:

5

Wyjście:

```

*
**
***
****
*****

```

Zadanie 11. squares – Kwadraty liczb naturalnych (0,5 pkt.)

Napisz program `squares`, który wczytuje ze standardowego wejścia liczbę naturalną n , po czym wypisuje na standardowe wyjście podzielne przez 2 lub przez 3 kwadraty liczb naturalnych mniejszych od n .

Przykładowe wykonanie

Wejście:

10

Wyjście:

4 9 16 36 64 81

Zadanie 12. meanmax – Średnia arytmetyczna liczb i największa z liczb (1 pkt.)

Napisz program `meanmax`, który czyta ze standardowego wejścia liczby rzeczywiste aż do napotkania znaku końca pliku i wypisuje na standardowe wyjście ich średnią arytmetyczną oraz największą z nich. Jeżeli nie wprowadzono żadnej liczby, program nic nie wypisuje.

Przykładowe wykonanie

Wejście:

2.5 0.3 1.7 0.25 3.75 [EOF]

Wyjście:

1.7 3.75

Zadanie 13. sum – Suma szeregu skończonego (1 pkt.)

Napisz program `sum`, który wczytuje ze standardowego wejścia liczbę naturalną n i wypisuje na standardowe wyjście sumę

$$4 \sum_{k=1}^n \frac{(-1)^{k+1}}{2k-1}.$$

Przykładowe wykonanie

Wejście:

1000

Wyjście:

3.14059

Zadanie 14. fibonacci – Ciąg Fibonacciego (1 pkt.)

Ciąg Fibonacciego zaczyna się od wyrazów 0 i 1, a każdy następny wyraz jest sumą dwóch poprzednich. Napisz program `fibonacci`, który wczytuje ze standardowego wejścia liczbę naturalną n i drukuje na standardowe wyjście n pierwszych wyrazów ciągu Fibonacciego.

Przykładowe wykonanie

Wejście:

10

Wyjście:

0 1 1 2 3 5 8 13 21 34

Zadanie 15. guess – Zabawa w zgadywanie liczby (1 pkt.)

Napisz program `guess` odgadujący pomyślaną przez użytkownika liczbę. Przed uruchomieniem programu użytkownik wybiera losowo liczbę całkowitą z przedziału od 0 włącznie do 100 wyłącznie. Po uruchomieniu programu wypisuje na standardowe wyjście pewną liczbę z tego przedziału i wczytuje ze standardowego wejścia odpowiedź użytkownika równą -1, 0, lub 1. Odpowiedzi te oznaczają odpowiednio, że pomyślana przez użytkownika liczba jest mniejsza, równa lub większa od liczby wyświetlonej przez program. Program kontynuuje zgadywanie aż do odgadnięcia właściwej liczby.

Zadanie 16. deserter – Brakująca liczba (1 pkt.)

Spśród liczb naturalnych od 1 do n włącznie usuwamy losowo jedną, a resztę zapisujemy w przypadkowej kolejności. Napisz program `deserter`, który czyta zapisane liczby ze standardowego wejścia aż do napotkania znaku końca pliku i wypisuje na standardowe wyjście tę brakującą.

Przykładowe wykonanie

Wejście:

2 5 1 4 [EOF]

Wyjście:

3

Zadanie 17. pass – Mijanie się zer i jedynek (1 pkt.)

Zapisujemy w jednej linii losowo zera i jedynki, oddzielając je spacjami. W pewnej chwili wszystkie zera przesuwać się na początek linii, a wszystkie jedynki na koniec. Napisz program `pass`, który czyta ze standardowego wejścia zapisane liczby aż do napotkania znaku końca pliku i wypisuje na standardowe wyjście, ile razy zera miną się z jedynekami.

Przykładowe wykonanie

Wejście:

0 1 1 0 1 0 1 [EOF]

Wyjście:

5

Zadanie 18. polygonal – Długość łamanej (1 pkt.)

Napisz program `polygonal`, który czyta ze standardowego wejścia współrzędne punktów na płaszczyźnie aż do napotkania znaku końca pliku i wypisuje na standardowe wyjście długość łamanej otwartej łączącej te punkty, od pierwszego do ostatniego. Jeżeli nie podano ani jednej pary liczb, program nic nie wypisuje.

Przykładowe wykonanie

Wejście:

-0.3 7.5

9.5 -3.7

5.0 0.4

[EOF]

Wyjście:

20.969

Zadanie 19. section – Losowy rozkład liczby (1 pkt.)

Napisz program `section`, który wczytuje ze standardowego wejścia dodatnią liczbę rzeczywistą x oraz liczbę naturalną n , po czym wypisuje na standardowe wyjście n losowych liczb dodatnich, których suma wynosi x .

Przykładowe wykonanie

Wejście:

10 3

Wyjście:

5.4095 3.45354 1.13696

Zadanie 20. pimc – Obliczanie liczby π metodą Monte Carlo (1 pkt.)

Przybliżoną wartość liczby π można wyznaczyć następująco. Rozważmy kwadrat o wierzchołkach w punktach $(0,0)$, $(0,1)$, $(1,0)$, $(1,1)$ oraz zawartą w nim ćwiartkę koła o środku w punkcie $(0,0)$ i promieniu 1. Pola kwadratu i ćwiartki koła wynoszą odpowiednio $A_s = 1$ i $A_c = \pi/4$. Wybierzmy losowo dużo punktów tak, aby równomiernie wypełniały cały kwadrat. Stosunek liczby N_c punktów wewnątrz ćwiartki koła do liczby N_s wszystkich punktów w kwadracie jest w przybliżeniu równy stosunkowi pól tych figur, $N_c/N_s \approx A_c/A_s$. Dostajemy stąd

$$\pi \approx \frac{4N_c}{N_s}$$

Napisz program `pimc`, który czyta ze standardowego wejścia liczbę punktów do wylosowania i wypisuje na standardowe wyjście otrzymane przybliżenie liczby π .

Przykładowe wykonanie

Wejście:

10000

Wyjście:

3.144

Zadanie 21. statistics – Średnia i odchylenie standardowe (0,5 pkt.)

Średnia arytmetyczna x z próby n liczb rzeczywistych x_1, \dots, x_n oraz jej odchylenie standardowe σ dane są wzorami

$$x = \frac{1}{n} \sum_{k=1}^n x_k, \quad \sigma = \sqrt{\frac{1}{n(n-1)} \sum_{k=1}^n (x_k - x)^2}$$

Napisz program `statistics`, który czyta ze standardowego wejścia liczby rzeczywiste aż do napotkania znaku końca pliku i wypisuje na standardowe wyjście ich średnią arytmetyczną oraz jej odchylenie standardowe.

Przykładowe wykonanie

Wejście:

```
1 2 3 4 5 6 7 8 9 10 [EOF]
```

Wyjście:

```
5.5 0.957427
```

Zadanie 22. `prime` – Czy liczba jest pierwsza? (1 pkt.)

Napisz funkcję `prime`, która przyjmuje jako argument liczbę naturalną i zwraca prawdę (`true`), jeśli jest ona pierwsza, lub fałsz (`false`) w przeciwnym razie. Korzystając z tej funkcji napisz program `prime`, który wczytuje ze standardowego wejścia liczbę naturalną i wypisuje na standardowe wyjście `Podana liczba jest pierwsza.`, jeśli jest ona pierwsza, lub `Podana liczba nie jest pierwsza.` w przeciwnym razie.

Przykładowe wykonanie

Wejście:

```
97
```

Wyjście:

```
Podana liczba jest pierwsza.
```

Zadanie 23. `bifactorial` – Podwójna silnia (1 pkt.)

Podwójna silnia liczby naturalnej n , oznaczana jako $n!!$, to iloczyn wszystkich liczb naturalnych mniejszych lub równych n o takiej samej parzystości jak n , przy czym $0!! = 1$. Napisz funkcję `bifactorial`, która przyjmuje jako argument liczbę naturalną i zwraca jej podwójną silnię. Korzystając z tej funkcji napisz program `bifactorial`, który wczytuje ze standardowego wejścia liczbę naturalną i wypisuje na standardowe wyjście jej podwójną silnię.

Przykładowe wykonanie

Wejście:

```
6
```

Wyjście:

```
48
```

Zadanie 24. `digits` – Cyfry dziesiętne (1 pkt.)

Napisz funkcję `digits`, która przyjmuje liczbę naturalną i zwraca liczbę cyfr w jej zapisie dziesiętnym, oraz funkcję `digit`, która przyjmuje liczbę naturalną oraz numer cyfry w jej zapisie dziesiętnym i zwraca tę cyfrę. Cyfry numerujemy od zera dla cyfry jedności. Korzystając z tych funkcji napisz program `digits`, który wczytuje ze standardowego wejścia liczbę naturalną i wypisuje na standardowe wyjście jej cyfry oddzielone spacjami.

Przykładowe wykonanie

Wejście:

```
517
```

Wyjście:

```
5 1 7
```

Zadanie 25. sqroot – Pierwiastek kwadratowy (1 pkt.)

Pierwiastek kwadratowy z liczby rzeczywistej x można obliczyć następująco. Jeżeli $x < 1$, pierwiastek leży między 0 a 1, zaś w przeciwnym razie między 1 a x . Bierzemy środek r odpowiedniego z tych przedziałów. Jeżeli $x < r^2$, to poszukiwany pierwiastek leży w lewej połowie przedziału, zaś w przeciwnym razie leży w prawej połowie. Do dalszych rozważań bierzemy więc odpowiednią połowę, dzielimy ją na pół i tak dalej. Dzięki temu w każdym kroku dwukrotnie zawężamy przedział, w którym leży pierwiastek. Ze względu na skończoną dokładność obliczeń środek któregoś kolejnego przedziału okaże się numerycznie równy jednemu z jego krańców.

Napisz funkcję `root`, która przyjmuje jako argument liczbę rzeczywistą i zwraca jej pierwiastek obliczony opisaną metodą. Korzystając z tej funkcji napisz program `root`, który wczytuje ze standardowego wejścia liczbę rzeczywistą i wypisuje na standardowe wyjście oddzielone spacją dwie wartości jej pierwiastka: pierwszą obliczoną za pomocą funkcji `root`, zaś drugą – odpowiednią funkcją biblioteczną.

Przykładowe wykonanie

Wejście:

7

Wyjście:

2.64575 2.64575

Zadanie 26. random – Liczby losowe (1 pkt.)

Napisz funkcję `random`, która przyjmuje jako argumenty liczby rzeczywiste a oraz b i zwraca losową liczbę rzeczywistą z przedziału $[a, b]$. Korzystając z tej funkcji napisz program `random`, który wczytuje ze standardowego wejścia liczby a oraz b , a także liczbę losowań n i wypisuje na standardowe wyjście n losowych liczb rzeczywistych z przedziału od a do b .

Przykładowe wykonanie

Wejście:

-5 5 7

Wyjście:

4.81355 -0.225038 2.74384 3.69302 -2.53199 -0.0405286 4.44732

Zadanie 27. coin – Rzut oszukaną monetą (1 pkt.)

Napisz funkcję `coin` symulującą rzut oszukaną monetą. Funkcja przyjmuje jako argument prawdopodobieństwo wyrzucenia orła i zwraca prawdę (`true`), jeśli wypadł orzeł, oraz fałsz (`false`), jeśli reszka. Korzystając z tej funkcji napisz program `coin`, który wczytuje ze standardowego wejścia prawdopodobieństwo wyrzucenia orła oraz liczbę rzutów i wypisuje na standardowe wyjście wyniki tych rzutów.

Przykładowe wykonanie

Wejście:

0.2 5

Wyjście:

orzeł reszka orzeł reszka reszka

Zadanie 28. rng – Generator liczb pseudolosowych (1 pkt.)

Napisz funkcję `rng` zwracającą pseudolosowe liczby całkowite, obliczane za pomocą prostego liniowego generatora kongruentnego: funkcja oblicza kolejną liczbę x_{n+1} na podstawie poprzedniej liczby x_n ze wzoru

$$x_{n+1} = (33x_n + 1) \bmod 1024.$$

Funkcja nie przyjmuje żadnych argumentów i nie korzysta ze zmiennych globalnych. Korzystając z tej funkcji napisz program `rng`, który wczytuje ze standardowego wejścia liczbę naturalną n i wypisuje na standardowe wyjście n kolejnych liczb pseudolosowych, poczynając od jedynki.

Przykładowe wykonanie

Wejście:

10

Wyjście:

1 34 99 196 325 486 679 904 137 426

Zadanie 29. euclid – Algorytm Euklidesa (1 pkt.)

Największy wspólny dzielnik dwóch liczb naturalnych można znaleźć następującą metodą, przypisywaną Euklidesowi. Pierwszą liczbę zastępujemy resztą z dzielenia przez drugą i zamieniamy liczby miejscami. Czynność tę powtarzamy aż jedna z liczb stanie się równa zeru. Wtedy druga jest poszukiwanym największym wspólnym dzielnikiem wyjściowych liczb. Najmniejszą wspólną wielokrotność dwóch liczb naturalnych można obliczyć dzieląc ich iloczyn przez największy wspólny dzielnik.

Napisz funkcję `euclid` obliczającą jednocześnie największy wspólny dzielnik i najmniejszą wspólną wielokrotność dwóch liczb naturalnych. Funkcja przyjmuje jako argumenty referencje dwóch zmiennych całkowitych, które przed wywołaniem funkcji zawierają wyjściowe liczby, i wpisuje do tych zmiennych odpowiednio największy wspólny dzielnik i największą wspólną wielokrotność tych liczb. Korzystając z tej funkcji napisz program `euclid`, który wczytuje ze standardowego wejścia dwie liczby naturalne i wypisuje na standardowe wyjście ich największy wspólny dzielnik oraz najmniejszą wspólną wielokrotność.

Przykładowe wykonanie

Wejście:

60 1240

Wyjście:

20 3720

Zadanie 30. quiz – Test z tabliczki mnożenia (1 pkt.)

Napisz program `quiz`, który wypisuje na standardowe wyjście pięć losowych pytań z tabliczki mnożenia liczb od 1 do 10 i wczytuje odpowiedzi ze standardowego wejścia. Każde pytanie zadaje aż do uzyskania poprawnej odpowiedzi.

Przykładowe wykonanie

Wyjście:

5 1

Wejście:

3

Wyjście:

5 1

Wejście:

5

Wyjście:

10 5

Wejście:

50

Wyjście:

3 5

Wejście:

```
15
Wyjście:
5 6
Wejście:
30
Wyjście:
8 7
Wejście:
56
```

Zadanie 31. factor – Rozkład liczby na czynniki pierwsze (1 pkt.)

Liczbę naturalną można rozłożyć na czynniki pierwsze w następujący sposób. Sprawdzamy, czy liczba jest podzielna przez 2. Jeżeli tak, dopisujemy 2 do rozkładu, a samą liczbę dzielimy przez 2. Czynność tę powtarzamy tak długo, aż liczba przestanie być podzielna przez 2. Następnie postępujemy w ten sam sposób, badając podzielność przez 3, 4 i tak dalej, aż rozważana liczba stanie się równa 1.

Napisz program `factor`, który wczytuje ze standardowego wejścia liczbę naturalną i wypisuje na standardowe wyjście jej rozkład na czynniki pierwsze.

Przykładowe wykonanie

```
Wejście:
517
Wyjście:
11 47
```

Zadanie 32. tri – Liczby trzycyfrowe (1 pkt.)

Napisz program `tri` wypisujący na standardowe wyjście w kolejności rosnącej wszystkie liczby trzycyfrowe, których cyfra setek to 1, 2, 5, 6, 7, lub 9, cyfra dziesiątek jest potęgą dwójki, cyfra jedności jest parzysta, a suma cyfr dzieli się przez 7. Program wypisuje te liczby przy użyciu odpowiednich pętli.

Wykonanie

```
Wyjście:
124 142 214 248 284 518 520 588 610 626 644 680 716 786 914 948 984
```

Zadanie 33. pitagoras – Trójki pitagorejskie (1 pkt.)

Trójką pitagorejską nazywamy każdą trójkę liczb naturalnych (a, b, c) taką, że $a^2 + b^2 = c^2$. Trójkę nazywamy pierwotną, jeżeli liczby a, b, c są względnie pierwsze.

Napisz program `pitagoras`, który wczytuje ze standardowego wejścia liczbę naturalną C i wypisuje na standardowe wyjście wszystkie pierwotne trójki pitagorejskie, dla których $c < C$.

Przykładowe wykonanie

```
Wejście:
30
Wyjście:
3 4 5
5 12 13
8 15 17
7 24 25
20 21 29
```