

### Zad. 1.

- Załóż katalog `programowanie` w którym będziesz pracować (`mkdir programowanie`) i przejdź do niego (`cd programowanie`)
- Za pomocą dowolnego edytora (np. Visual Studio) przepisz (nie kopiuj z pliku pdf) program jak poniżej:

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

int main{

    cout << "Podaj liczbe całkowita= " << endl
    int liczba1
    cin >> liczba1;

    cout << Podaj liczbe rzeczywista=  << endl;
    double liczba2;
    cin >> liczba2

    wynik=liczba1/liczba2

    cout << "wynik= " < wynik << endl;

    return 0;
}
```

- Nadaj plikowi nazwę np. `prog.cpp`
- Skompiluj program za pomocą polecenia `g++` (`g++ prog.cpp`)
- Do kompilacji dodaj flagę `-Wall`
- Popraw błędy i uruchom program
- Dodatkowo: sprawdź sam etap preprocesora i kompilacji (patrz - materiał wykładowy). Opcje polecenia `g++` można znaleźć pisząc `man g++`. Jakie pliki powstają we wszystkich trzech przypadkach ?

### Zad. 2.    Napisz program który:

1. wypisze na ekran:  
*Podaj dwie liczby całkowite*
2. wczyta te dwie liczby;
3. obliczy ich iloraz, resztę z dzielenia pierwszej liczby przez drugą, średnią arytmetyczną oraz geometryczną;
4. przed próbą dzielenia liczb przez siebie i liczeniem pierwiastka sprawdza poprawność wprowadzonych danych i w przypadku dzielenia przez zero lub pierwiastka z liczby ujemnej przerywa działanie programu (`return 1;`);
5. wypisze na ekran: `a/b = dd`  
`reszta z dzielenia a/b = rr`  
`Średnia arytmetyczna a i b = xx`  
`Średnia geometryczna a i b = yy`

Napisy `dd`, `rr`, `xx` i `yy` muszą być zastąpione wynikami obliczeń.

Do policzenia średniej geometrycznej użyj funkcji `sqrt` z biblioteki `cmath` (konieczne jest dodanie na początku pliku linii `#include<cmath>`).

Operator reszty z dzielenia to `%`

Przykładowo: `15%4` to reszta z dzielenia 15 przez 4.

**Zad. 3.** Napisz program znajdujący największą liczbę całkowitą spośród trzech podanych z klawiatury ( i wypisujący na ekran jej wartość). Uwaga: spróbuj napisać jak najkrótszy algorytm do znajdowania największej liczby spośród podanych trzech ( użyj tylko dwóch instrukcji warunkowych `if` bez `else`, bez `&&` i bez `||`).

**Zad. 4.** Napisz program szukający pierwiastków równania kwadratowego. Program powinien:

1. wypisywać na ekran komunikat:  
`Podaj współczynniki równania kwadratowego`  
a następnie wypisywać:  
`Podaj a`  
i wczytywać z klawiatury `a`.  
Analogicznie dla `b` i `c`.
2. Powinna być tworzona nowa zmienna `delta` i pod nią podstawiona wartość wyliczonej  $\Delta$  równania kwadratowego. **Wypisz na ekran wartość obliczonej delty.**

3. Następnie sprawdzane powinny być trzy warunki : ( $\Delta < 0$ ,  $\Delta = 0$  i  $\Delta > 0$ ) i w zależności od wartości  $\Delta$  program powinien wypisywać na ekran wartości pierwiastków z komentarzem: Rownanie nie ma rzeczywistych pierwiastkow, Rownanie ma jeden pierwiastek = ... lub Rownanie ma dwa pierwiastki:  $x_1 = \dots$  i  $x_2 = \dots$ .

**Ważne – sprawdź czy program prawidłowo działa i znajduje pojedynczy pierwiastek dla  $a=1$ ,  $b=-0.2$  i  $c=0.01$**

**Zad. 5.** Napisz program, który wczytuje ze standardowego wejścia rok, a następnie wypisuje na standardowe wyjście czy ten rok jest przestępny czy nie. Na początku powinno być sprawdzane, czy nie podaliśmy liczby mniejszej lub równej zero. W przypadku podania błędnych danych program powinien przerywać działanie z odpowiednim komunikatem (użyj polecenia `return`). Rok przestępny spełnia jeden z warunków:

- jest podzielny przez 4 z wyjątkiem lat podzielnych przez 100
- jest podzielny przez 400

**Zad. 6.** Wypisz na dwa sposoby na ekran liczby od 0 do  $n$ .  $n$  powinno być wczytywane z klawiatury. Skorzystaj z pętli `for` i `while` lub `do-while`. Obie pętle powinny znajdować się w jednym programie.

**Zad. 7.** Napisz program, który będzie wczytywał dwie liczby całkowite będące dolną i górną granicą przedziału liczb całkowitych i będzie zliczał i wypisywał na ekran ile liczb parzystych znajduje się pomiędzy wczytanymi liczbami. Po wczytaniu granic program powinien sprawdzać czy dolna granica jest mniejsza od górnej, a jeśli nie to powinien wypisywać komunikat, że jest problem i przerywać działanie za pomocą polecenia:

`return 1;`

**Zad. 8.** Napisz program sumujący kwadraty kolejnych  $n$  liczb naturalnych i wypisujący wynik na ekran.  $n$  podawane jest z klawiatury.

**Zad. 9.** Napisz program do liczenia  $n!$  Porównaj dla jak dużych liczb wynik ma sens w przypadku operowania na liczbach całkowitych i zmiennoprzecinkowych.

**Zad. 10.** Użyj **jednej** pętli `while` do wypisania na ekran obok siebie kolumn dwóch liczb: pierwszej z liczbami od 0 do  $n$  i drugiej z liczbami od  $n$  do 0.  $n$  powinno być wczytywane z klawiatury. Spróbuj stworzyć jak najkrótszy algorytm.

**Zad. 11.** Napisz program, który będzie prosił o podanie ze standardowego strumienia wejściowego dwóch liczb rzeczywistych i obliczał ich iloraz. Jeżeli

liczba przez którą dzielimy jest równa zero, powinien prosić o podanie innej pary liczb i tak do skutku. Użyj nieskończonej pętli `while` i instrukcji `continue` i `break`.