

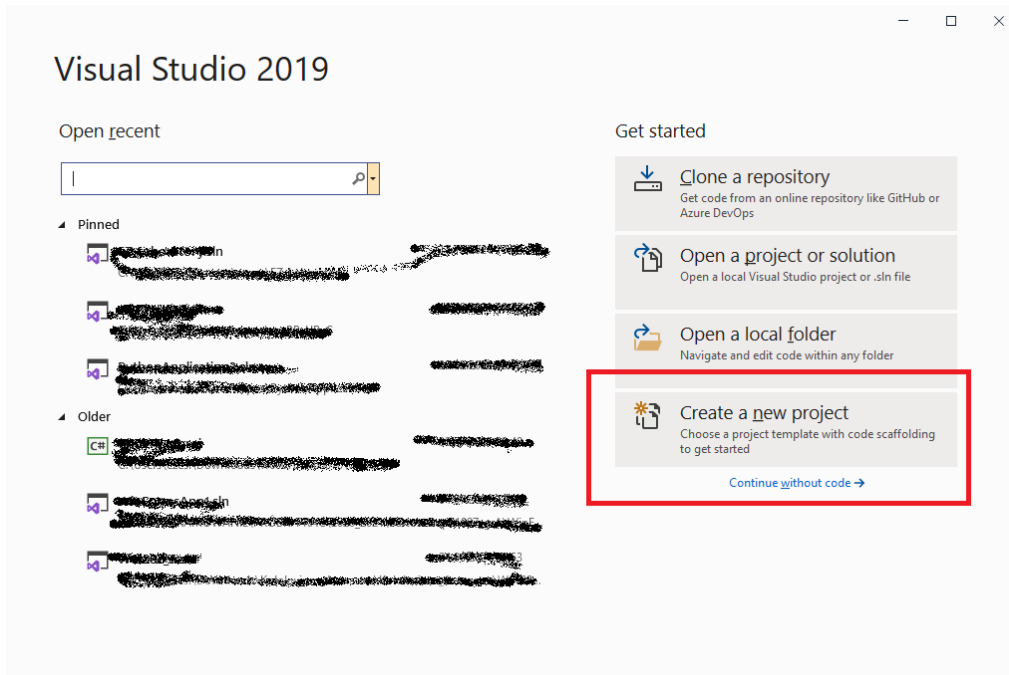
# Podstawy Programowania w języku C

## Laboratorium 1

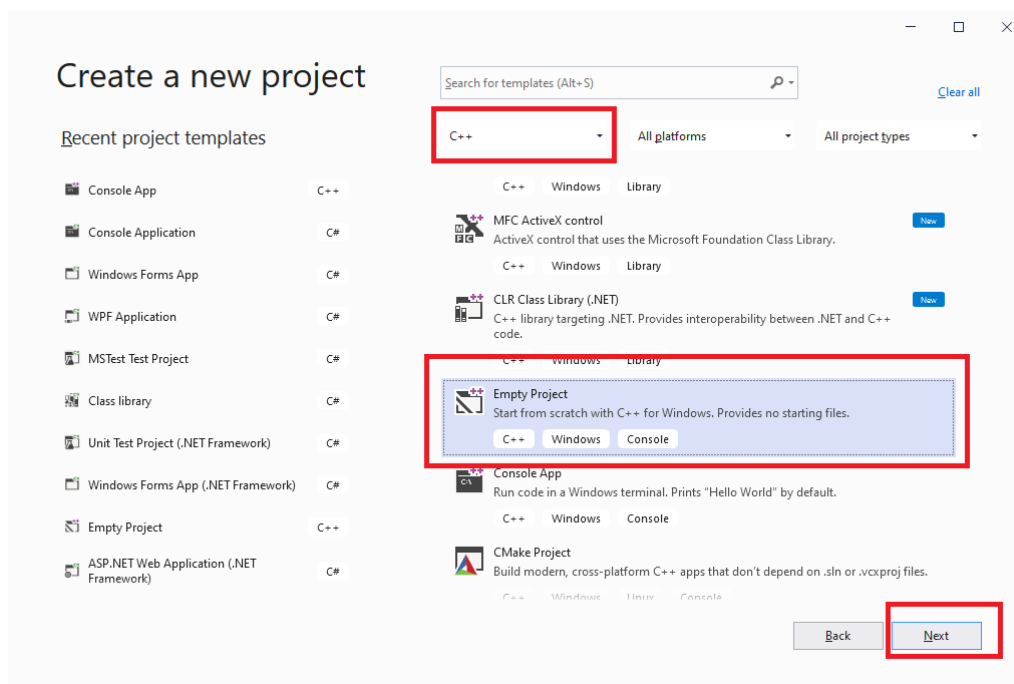
### DEFINIOWANIE ZMIENNYCH, TYPY DANYCH, INSTRUKCJE WARUNKOWE

#### Tworzenie projektu - Visual Studio (VS)

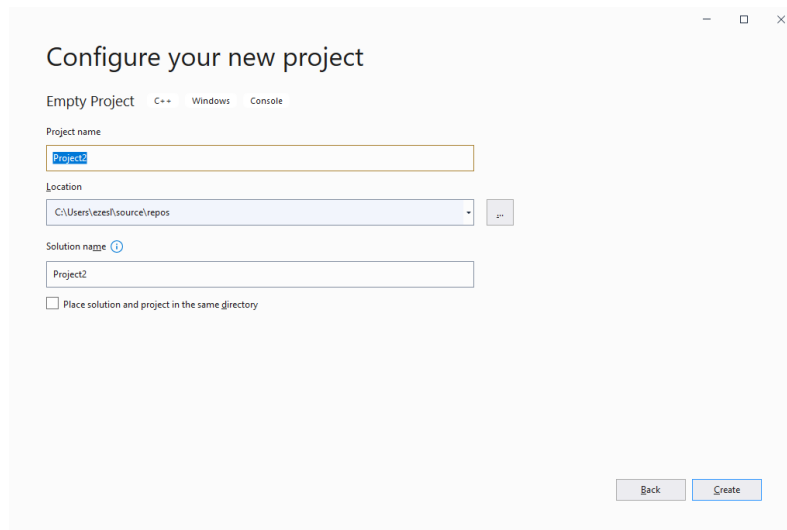
Należy otworzyć VS i kliknąć Create a new project



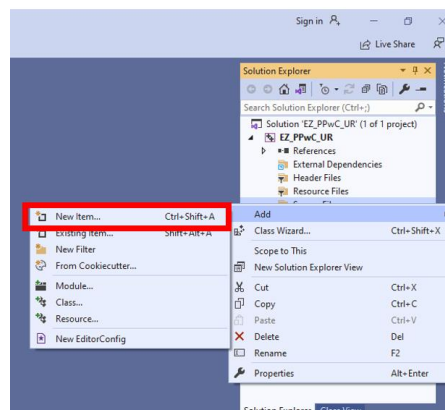
W kolejnym kroku należy wybrać język C++ i zaznaczyć Empty Project i kliknąć Next



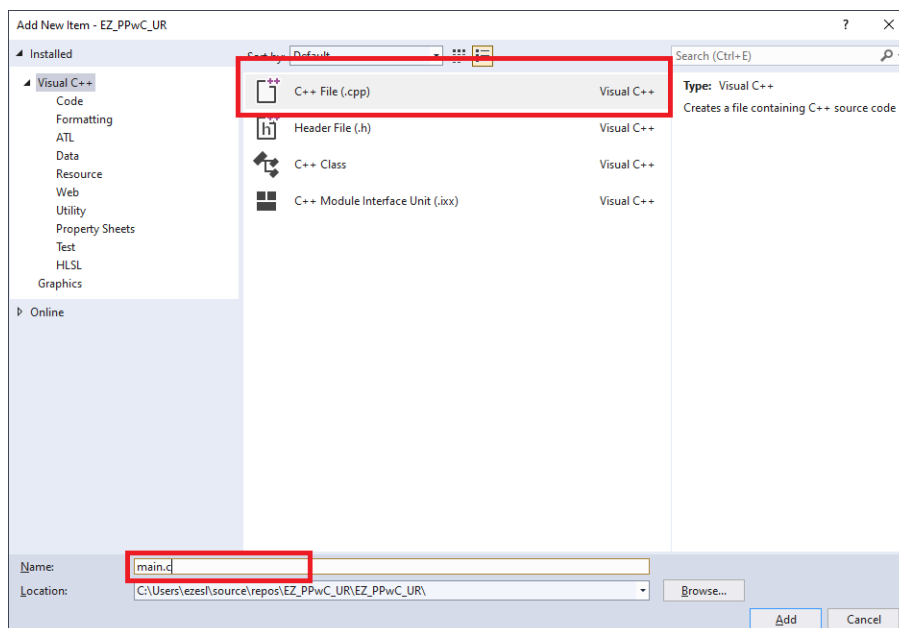
Wpisać odpowiednio nazwę i kliknąć Create



W kolejnym kroku w zakładce Solution Explorer, klikamy prawym przypiskiem myszy na Source File i klikamy New Item



Należy wybrać stworzenie pliku C++ i wpisać nazwę np.: main.c i kliknąć Add. **UWAGA!!! rozszerzenie pliku musi być zdefiniowane jako \*.c.**



## Pierwszy program

```
#include <stdio.h> // biblioteka wejścia i wyjścia

int main() {
    printf("Hello world!\n"); // strumień wyjścia, wyświetlania danych na konsoli

    return 0; // funkcja kończąca działanie programu
}
```

## Deklaracja zmiennych

Przy deklaracji zmiennej należy określić jej typ oraz nazwę zmiennej. Identyfikatory typu są koniecznymi argumentami dla funkcji `printf()` i `scanf()`. Poniżej przedstawiono różne typy i odpowiadające im identyfikatory typu:

- liczby całkowite: `int`, `%d`
- liczby zmiennoprzecinkowe: `float` (4 bajty), `%f`
- liczby zmiennoprzecinkowe: `double` (8 bajtów), `%lf`
- znaki: `char`, `%c` i `%s`

```
// deklaracja zmiennych wraz z inicjalizacją
int a, b=2;
float d=1.2;
double doubleA=2.45;
char znak = 'z';
```

## Funkcja `printf()` i `scanf()`

Funkcja `printf()` służy do wyświetlania tekstu i wartości zmiennych, poniżej kilka przykładów użycia:

- `printf("podaj liczbę: ")` - funkcja przyjmuje jeden argument tekstowy, tekst powinien być zapisany w cudzysłowie "...",
- `printf("%d", liczba)` - funkcja przyjmuje dwa argumenty, pierwszym z nich jest identyfikator typu zmiennej, którą chcemy wyświetlić (`%d`), natomiast drugim argumentem jest nazwa zmiennej (`liczba`). Argumenty są podawane po przecinku. Identyfikator typu informuje program w którym miejscu ma być wyświetlona zmienna i jakiego jest typu. Funkcja `printf()` może wyświetlać więcej niż jedną zmienną, przykład wyświetlenia dwóch zmiennych (które przed wywołaniem funkcji muszą być zadeklarowane): `printf("jedna zmienna %d, druga zmienna %d", a, b);`

Funkcja `scanf()` służy do wprowadzania tekstu z klawiatury i przypisywania go do zmiennej, poniżej przykład użycia:

- `scanf("%d", &liczba)` - funkcja przyjmuje dwa argumenty, pierwszym argumentem jest identyfikator typu, a drugim argumentem funkcji (po przecinku) jest nazwa zmienna poprzedzona symbolem `&`.

Zwróć uwagę na fakt, że obie funkcje przyjmują 2 argumenty: identyfikator typu oraz nazwę zmiennej. Jednak funkcja `scanf()` do prawidłowego działania potrzebuje wprowadzenia symbolu `&` przed nazwą zmiennej. Konieczność wprowadzenia tego symbolu wyjaśnimy sobie na zajęciach ze wskaźnikami.

- `scanf("%d", &a);` // wczytaj z klawiatury zmienną typu `int` i przypisz do zmiennej `a`
- `printf("%d", a);` // wyświetl zmienną `a`, która jest typu `int`

Przykład wyświetlania i wczytywania danych z klawiatury:

```
#include <stdio.h>

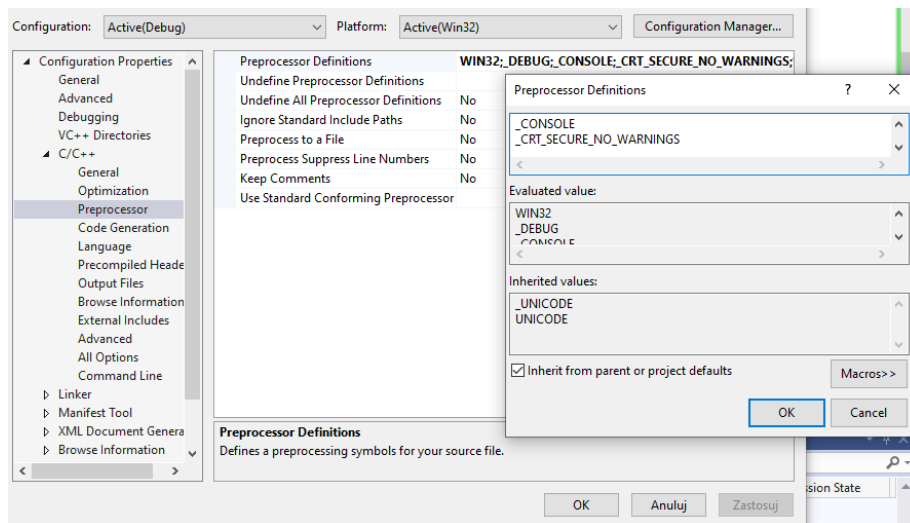
int main()
{
    //deklaracja zmiennych wraz z inicjalizacja
    int a, b=2;
    float d=1.2;
    double doubleA=2.45;
    char znak = 'b';

    printf("Liczba całkowita: %d", b);
    printf("\nLiczba float: %f", d);
    printf("\nLiczba double: %lf", doubleA);
    printf("\nZnak: %c", znak);

    //pobranie liczby z klawiatury
    printf("\nPodaj liczbę: ");
    scanf("%d", &a); // wczytanie liczby z klawiatury
    printf("Wczytana liczba %d: ", a);

    return 0;
}
```

W przypadku pojawienia się błędu Error C4996 'scanf': This function or variable may be unsafe. Consider using scanf\_s instead. To disable deprecation, use \_CRT\_SECURE\_NO\_WARNINGS. See online help for details. należy dodać \_CRT\_SECURE\_NO\_WARNINGS do Preprocessor Definitions (Project -> Properties).



Operatory matematyczne: + - \* / %

Operatory porównania: > < >= <= != ==

Operatory logiczne: AND - &&, OR - ||, NOT - !

### Operator sizeof

Operator sizeof zwraca rozmiar w bajtach (gdzie bajtem jest zmienna typu char) podanego typu lub typu podanego wyrażenia. Ma on dwa rodzaje: sizeof(typ) lub sizeof wyrażenie. Przykładowo:

```
printf("sizeof(short ) = %d\n", sizeof(short));
printf("sizeof(int ) = %d\n", sizeof(int));
printf("sizeof(long ) = %d\n", sizeof(long));
```

```
printf("sizeof(float ) = %d\n", sizeof(float));  
printf("sizeof(double) = %d\n", sizeof(double));
```

## Instrukcje sterujące:

### Instrukcja if

```
if (wyrażenie) {  
    /* blok wykonany, jeśli wyrażenie jest prawdziwe */  
}  
/* dalsze instrukcje */  
  
if (wyrażenie) {  
    /* blok wykonany, jeśli wyrażenie jest prawdziwe */  
}  
else {  
    /* blok wykonany, jeśli wyrażenie jest nieprawdziwe */  
}  
/* dalsze instrukcje */
```

### Instrukcja switch

```
switch (wyrażenie) {  
    case wartość1: /* instrukcje, jeśli wyrażenie == wartość1 */  
        break;  
    case wartość2: /* instrukcje, jeśli wyrażenie == wartość2 */  
        break;  
    /* ... */  
    default: /* instrukcje, jeśli żaden z wcześniejszych warunków */  
        break; /* nie został spełniony */  
}
```

## Zadania do samodzielnego rozwiązania:

1. Napisać program, który wypisuje na konsolę nr albumu studenta (osoby rozwiązującej program).
2. Napisać program, który sprawdza, czy podana liczba całkowita, wczytana z klawiatury jest parzysta, czy nieparzysta. (operator %).
3. Napisać program, który pozwala użytkownikowi na wprowadzenie dwóch liczb zmiennoprzecinkowych, a następnie zwraca informację, która z wprowadzonych liczb jest liczbą większą.
4. Napisać program, który pozwala użytkownikowi na wprowadzenie czterech liczb zmiennoprzecinkowych, a następnie zwraca średnią arytmetyczną z wprowadzonych liczb.
5. Napisać prosty kalkulator wykonujący działanie matematyczne (+, -, /, \*) na dwóch liczbach podanych z klawiatury, wykorzystując instrukcję switch.