

1. Napisz program, który będzie wyliczał sumę kolejnych liczb całkowitych od 1 do n, gdzie n ma być wczytane z klawiatury. Zrób to na dwa sposoby:
 - a. za pomocą wzoru $n(n+1)/2$,
 - b. za pomocą zmiennej pomocniczej sumaLiczb i instrukcji for.
2. Zmodyfikuj program z podpunktu b, aby wyliczał silnię (iloczyn kolejnych liczb).
3. Za pomocą instrukcji while napisz program do liczenia silni.
4. Za pomocą instrukcji while wypisz liczby od 1 do n. N wczytaj z klawiatury. (porównaj kod z użyciem instrukcji for).
5. Z użyciem pętli do while napisz program który będzie prosił o wpisanie liczby ujemnej. Jeśli będzie wprowadzona poprawna liczba, ma się wyświetlić komunikat „Brawo – poprawna liczba”, jeśli liczba jest błędna, należy poprosić o podanie liczby jeszcze raz.
6. Napisz program który będzie prosił o kod dostępu dopóki nie dostanie poprawnego. Gdy go otrzyma wyświetli napis "Dostęp przyznany!".
7. Napisz program, który będzie porównywał zgodność dwóch wprowadzonych haseł. Jeśli jest błąd – ma wyświetlić się informacja o błędzie i poprosić użytkownika o ponowienie próby. Jeśli jest poprawna – ma być informacja o zgodności haseł.
8. Napisz program, który poprosi użytkownika o podanie 3 liczb. Po ich podaniu na ekranie ma się wyświetlić największa z nich.
9. Napisz program, który poprosi o podanie liczby. Jeśli liczba jest podzielna przez 5, ma wyświetlić się napis tak. W przeciwnym wypadku nie.

10. Napisz program – mini grę „Arena”. Pracę wykonaj etapami:
- a. pobierz od użytkownika nazwy dwóch bohaterów i zapisz je jako zmienne typu string np. bohater1 i bohater2,
 - b. stwórz dwie zmienne hp1 i hp2 typu int (mają przechowywać ilość punktów życia), ustaw ich wartości na 50.
 - c. dodaj instrukcję do while, która będzie prosiła użytkownika o wprowadzenie litery (typ string, nazwa zmiennej - akcja), jeśli użytkownik wprowadzi x (małe) to ma wyświetlać się napis „koniec gry”, jeśli jest inny znak – ma prosić ponownie o wprowadzenie litery (póki co dopuszczamy wprowadzanie tylko małych liter – ale w programie nie będziemy tego sprawdzać),
 - d. zmodyfikuj wewnątrz instrukcji do, tak, aby po wprowadzeniu litery a (małe) hp1 rosło o 5, a hp2 malało o 5 (instrukcja if).
 - e. po napisie koniec gry wyświetl na ekranie tablice wyników (wyświetl nazwy obu bohaterów) oraz aktualne wartości ich pól hp1 i hp2.
 - f. przed liniijką wyświetlającą napis koniec gry dodaj liniijkę czyszczącą konsolę (system("cls");) – do początku pliku dodaj `#include <cstdlib>`
 - g. Po tablicy wyników wyświetl informację kto jest zwycięzcą (osoba mająca większą ilość punktów życia) – instrukcja if else (uwzględnij remis).
 - h. rozbuduj warunek w while tak aby była wykonywana pętla kiedy hp1 i hp2 jest większe lub równe 0 (użyj &&)
 - i. jak w podpunkcie d dodaj dodatkowe operacje dla wpisania innych liter.
 - j. dodaj opcje else przy wprowadzeniu błędnego znaku