

Wstęp do programowania. Lista 6. Pętla *while*.

1. Napisz program, który sprawdzi czy podana przez użytkownika liczba jest kwadratem liczby naturalnej. Użyj pętli *while*.
2. Napisz program, który znajdzie i wyświetli największe n o własności $2^n \leq k$, dla dowolnej wczytanej z klawiatury liczby naturalnej k . Użyj pętli *while*.
3. Napisz program, który wyświetli najmniejszą liczbę naturalną n taką, że $\frac{1}{n} < 0.0043212$. Użyj pętli *while*.
4. Napisz program, który wyświetli najmniejszą liczbę naturalną n taką, że $1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{n} > 11$. Użyj pętli *while*.
5. Napisz program, który do listy $l = [a']$ będzie dodawał kolejne znaki wczytywane z klawiatury, do momentu, aż wpisujemy *'stop'*. Na koniec program ma wyświetlić naszą listę.
6. Napisz program, który wyznaczy NWD z dwóch podanych liczb. (patrz plik *nwd.pdf*)
7. Napisz program, który poprosi użytkownika o podanie dwóch wyrazów tej samej długości, następnie wyświetli łańcuch znaków powstały przez naprzemienne wypisanie kolejnych znaków z podanych wyrazów (po wpisaniu *kot* i *rak* program powinien zwrócić *kroatk*). Program powinien działać do momentu podania wyrazów równej długości.
8. Napisz program, który wylosuje pewną liczbę naturalną od 13 do 100. Następnie użytkownik ma zgadnąć jaka to liczba. W momencie podania prawidłowej liczby program ma wyświetlić gratulację i liczbę prób, w przypadku podania *'q'* program ma zakończyć działanie informując użytkownika jaka była liczba szukana i ile prób wykonał użytkownik.
9. Napisz program, który pyta użytkownika o wynik dodawania dwóch liczb liczb całkowitych z przedziału $(100, 1000)$. Program ma działać do momentu podania trzech prawidłowych odpowiedzi pod rząd.
10. Napisz program, który będzie dodawał nowe pozycje w słowniku $\{ 'produkt': cena \}$ lub aktualizował cenę w przypadku produktu, który jest już w słowniku. Program ma działać, do momentu podania *quit*.
11. Program ma losować orzeł/reszka nie podając wyniku losowania. Gracz ma zgadywać wynik tego losowania. Jeśli zgadnie wygrywa 9 słoneczek, jeśli przegra traci 10 słoneczek. Użytkownik zaczyna grę mając 100 słoneczek, program ma działać do momentu gdy gracza nie stać na kolejną rundę lub podwoił liczbę słoneczek.
12. Napisz program, który wypisze n pierwszych trójek pitagorejskich postaci (a, b, c) , gdzie $0 < a \leq b \leq c \leq 100$, n podaje użytkownik.
13. Napisz program, będący grą w oczko.