

1. Домашна работа

1 Списък

Напишете програма в която потребителя въвежда числа в списък със следните условия:

Когато се въведе Add [NUMBER] се добавя число в списъка.

Когато се въведе Remove [NUMBER_REMOVE] се премахва това число от списъка.

Ако това число не съществува в списъка се намира най – близкото по – малко число до [NUMBER_REMOVE] и се премахва.

Когато потребителя въведе “end” се отпечатват всички числа от списъка.

Вход	Изход
Add 10 Add 20 Remove 12 end	20
Add 30 Add 20 Add 25 Remove 25 end	30 20

2 Речник

Пешо решил да криптира съобщенията си, защото всички социални мрежи се следят. Той си направил следния шифър:

A – 65

B – 66

C – 67

D – 68

...

Y – 89

Z – 90

Всяка буква отговаряла на число. Направете програма, която при въвеждане на текст преобразува текста в числа и обратно като така ще помогнете на Пешо да изпраща тайни съобщения и да ги декриптира.

Бележка: Съставете речник от ключ *string* и стойност *число*

Вход	Изход
Testing	84 69 83 84 73 78 71
This is secret	84 104 105 115 32 105 115 32 115 101 99 114 101 116

1. Домашна работа

3 Bank Account

Създайте клас **BankAccount** с поле **balance** от тип **decimal**.
Създайте методите **Deposit(decimal money)** и **Withdraw(decimal money)** и **ShowBalance()**.
Метода **Deposit** добавя пари към **balance**.
Метода **Withdraw** изважда пари от баланса, ако има достатъчно.
Метода **ShowBalance** показва баланса.
При правилно създаден клас трябва да можете да правите следното:
BankAccount ba = **new BankAccount()**;
ba.**Deposit**(30000);
ba.**Withdraw**(1000);
ba.**ShowBalance()**; //Output: 29000

4 Триъгълник

Създайте клас **Triangle** със страни **a,b,c** и метода **Perimeter()**. Потребителя въвежда страните на триъгълника в класа, след което се извиква метода **Perimeter** и програмата отпечатва периметъра на триъгълника.
Triangle t = **new Triangle()**;
t.a = Вход от конзолата 5
t.b = Вход от конзолата 6
t.c = Вход от конзолата 7
t.**Perimeter()**; //Output: 18

5 Студенти

Създайте клас **Student** съдържащ име на студента и среден успех.
Съставете програма в която потребителя въвежда **N** пъти информация за студент в формата {Име}, {Оценка}.
Данните трябва да се съхраняват в **List**.
Когато потребителя въведе **end** програмата показва всички студенти подредени по успех и средното аритметично.

Вход	Изход
Pesho,4 Tosho,5 Gosho,6	Gosho – 6 Pesho – 5 Tosho – 4 Average: 5