Exercício 1:

#include<iostream>

#include<stdlib.h>

#include<time.h>

#include<stdio.h>

using namespace std;

void preencheVetor(int v[], int tam){

    srand(time(NULL));

    for(int i = 0; i < tam; i++){

        v[i] = rand()%10 + 100;*//100...110*

    }

}

void mostraVetor(int v[], int tam){

    for(int i = 0; i < tam; i++){

        cout << " | " << v[i];

    }

    cout << "|" << endl;

}

bool testa(int vet[], int tam, int x){

    for(int i = 0; i < tam; i++){

        if(vet[i] == x){

            return true;

        }

    }

    return false;

}

int main(){

    int vetor[100], num, tam;

    cout<<"Digite o tamanho do vetor: "<<endl;

    cin >> tam;

    cout<<"Digite um numero: "<<endl;

    cin >> num;

    preencheVetor(vetor, tam);

    mostraVetor(vetor, tam);

    if(testa(vetor, tam, num) == true){

        cout<<"Pertence!"<<endl;

    } else{

        cout<<"Nao Pertence!"<<endl;

    }

}

Exercício 2:

#include<iostream>

#include<stdlib.h>

#include<time.h>

#include<stdio.h>

#define TAM 3

using namespace std;

int main(){

    int vet[TAM], i, maior, menor;

*//preenche o vetor*

    for(i = 0; i < TAM; i++){

        scanf("%d", &vet[i]);

    }

*//determina o maior valor*

    maior = vet[0];

    for(i = 1; i < TAM; i++){

        if(vet[i] > maior){

            maior = vet[i];

        }

    }

*//determina o menor valor*

    menor = vet[0];

    for(i = 1; i < TAM; i++){

        if(vet[i] < menor){

            menor = vet[i];

        }

    }

*//mostra o vetor*

    printf("Vetor: \n");

    for(i = 0; i < TAM; i++){

        printf("%d ", vet[i]);

    }

*//mostra o maior*

    printf("\nO maior valor do vetor eh: %d\n", maior);

*//mostra o menor*

    printf("O menor valor do vetor eh: %d\n", menor);

}

Exercício 3: O código retorna “true” se a string digitada possuir as vogais a, e, i, o, u, e “false” se não possuir.

Exercício Resolvido 1:

boolean isConsoante(String s, int i){

    boolean resp = true;

    if(i == s.lenght()){

        resp = true;

    }else if (isConsoante(s.charAt(i)) == false){

        resp = false;

    }else{

        resp = isConsoante(s, i + 1);

    }

    return resp;

}

Exercício 4: A versão da esquerda é a que possui o código mais simplificado e mais fácil de se entender entre as duas.

Exercício 6: O método que possui “return i--“ decrementará o valor de i, mas retornará o valor original que era i mantido antes de ser decrementado. Já o método que possui “return --i” decrementará o valor de i e retornará o valor decrementado.

Exercício 7: Enquanto a condição for verdadeira, o programa ira incrementar uma unidade a cada tipo primitivo.

Exercício Resolvido 2: O código imprime 46 – 11 pois os operadores shift right e left (>> e <<) deslocam os bits para a direita e esquerda e inserem um zero na posição vazia. Isso fez com que o número 10111 (23 representado pelo sistema binário), que foi declarado para as variáveis x e y, se transformasse em 1011 (11 representado pelo sistema binário) e em 101110 (46 representado pelo sistema binário), imprimindo assim 46 – 11 na tela.