



Universidade Federal do Espírito Santo – Centro Tecnológico
Departamento de Informática

Prof. Flávio Miguel Varejão

Instruções gerais para o BOCA: O BOCA é um programa de correção automática dos exercícios. Portanto, é necessário seguir estritamente os padrões de entrada e saída das questões. Apesar de em um programa comum para um usuário qualquer você ter que usar `printf's` para pedir uma entrada de dados para o usuário, em programas para o BOCA os `printf's` e `scanf's` devem ser usados com muita cautela, e seguindo rigorosamente os formatos de entrada e saída definidos para cada questão a ser submetida para o BOCA. O sistema compara letra por letra da saída do seu programa (isto é, tudo que foi escrito na tela) com a saída esperada por ele, portanto tudo deve ser escrito na saída padrão (ex. tela do monitor) conforme indicado nos exemplos das questões. Qualquer `printf` realizado sem necessidade pode invalidar a resposta. Lembre-se que os exemplos dados podem não cobrir todos os casos de teste das questões.

(BOCA:AL4_1) Problema: Dado o programa abaixo que lê opções de operações a serem realizadas e suas respectivas entradas e apresenta, de acordo com as escolhas feitas, o enésimo menor número primo, o enésimo termo da série de Flavionacci, a enésima potência de um número natural e o enésimo menor múltiplo comum de dois números naturais, implemente as funções `enesimoPrimo`, `enesimoTermo`, `enesimaPotencia` e `enesimoMultiploComum` (conforme cabeçalhos definidos abaixo) que fazem respectivamente cada uma das operações disponíveis. As únicas alterações permitidas no programa são a implementação dessas funções e eventuais implementações de funções auxiliares para a implementação das funções correspondentes às operações. A série de Flavionacci começa com os números naturais 0 1 2 e cada novo termo corresponde a multiplicação dos dois anteriores subtraída do terceiro antecessor. Eis um exemplo da série com 8 termos: 0 1 2 2 3 4 10 37.

Entrada: Sua entrada deverá conter uma sequência de números naturais indicando as operações que devem ser realizadas e os respectivos argumentos dessas operações. Atenção que a sequência sempre se encerra com o natural zero.

Saída: Sua saída deverá conter em sequência os resultados das operações solicitadas.

Exemplo de Entrada:

```
1 3 2 5 3 2 3 4 5 2 3 0
2 7 0
0
```

3 3 2 1 7 0
4 5 1 8 3 2 2 1 4 0

Exemplo de Saída:

5 3 8 30
10

27 17
40 4 7

=====

```
#include <stdio.h>
```

```
int enesimoPrimo(int n) {}
```

```
int enesimoTermo(int n) {}
```

```
int enesimaPotencia(int m, int n) {}
```

```
int enesimoMultiploComum(int n, int num1, int num2) {}
```

```
int main(){  
    int opc, n, num1, num2, result;
```

```
    scanf ("%d", &opc);
```

```
    while (opc){
```

```
        switch (opc) {
```

```
            case 1:
```

```
                scanf ("%d", &n);
```

```
                result = enesimoPrimo(n);
```

```
                printf(" %d\n", result);
```

```
                break;
```

```
            case 2:
```

```
                scanf ("%d", &n);
```

```
                result = enesimoTermo(n);
```

```
                printf(" %d\n", result);
```

```
                break;
```

```
            case 3:
```

```
                scanf ("%d%d", &num1, &num2);
```

```
                result = enesimaPotencia(num1, num2);
```

```
                printf(" %d\n", result);
```

```
                break;
```

```
            case 4:
```

```
                scanf ("%d%d%d", &n, &num1, &num2);
```

```
                result = enesimoMultiploComum(n, num1, num2);
```

```
                printf(" %d\n", result);
```

```
        break;
    default:
        printf("Opcao Invalida! ");
    }
    scanf ("%d", &opc);
}
return 0;
}
```