

Nome completo: _____ Data: 15/05/2014
Matrícula: _____ Professor: () Rafael () Fernando

1. A prova pode ser feita a lápis, porém o professor se dará ao direito de não aceitar reclamações relativas à correção.
2. Início da prova 07h30, término 10h00. Manter celulares desligados!
3. Coloque o seu nome nas folhas respostas.
4. Não é permitido o uso de calculadora ou similares.
5. **A compreensão das questões faz parte da prova.**

Boa prova!

1) (2.0) Dado o código à esquerda, preencha a tabela à direita com todas as mudanças de valor das variáveis até o final da execução do programa. A tabela está preenchida até a execução da primeira linha (`int i = 0, j = 1, a = 0, b = 2;`).

```
int main() {
    int i = 0, j = 1, a = 0, b = 2;
    int c, d;

    for(i = 2; i > 0; i--) {
        j = 0;

        while(i > 1 && j < 2) {
            a = b - 1;
            j++;
            b += 2;
        }

        d = 0;
        do {
            d++;
            j = d + 1;
        } while(d == 1 || d == -1);

        if(d == 1) {
            c = 10;
            d = c % 3;
        }

        if((i + j) + 1 == 5) {
            c = -2;
        } else {
            c = -3;
        }
    }

    return 0;
}
```

[illegible]

2) (3.0) Faça um programa em linguagem C que receba dois números inteiros: o primeiro corresponde ao N-ésimo número primo a ser mostrado, o segundo informa quantos números primos serão mostrados depois desse. Um número primo é aquele que somente é divisível por 1 e por ele mesmo. Exemplos de números primos: 2, 3, 5, 7, etc. Se o N-ésimo informado for 4, por exemplo, o usuário quer o 4º número primo existente (que é o número 5). Abaixo estão alguns exemplos:

```
Ex.: Entrada: 1 e 3, saída: 1 2 3 5
Ex.: Entrada: 6 e 2, saída: 11 13 17
Ex.: Entrada: 5 e 1, saída: 7 11
Ex.: Entrada: 2 e 5, saída: 2 3 5 7 11 13
```

3) (2.0) Dado o código em C à esquerda, mostre no espaço à direita (que simboliza a tela) o que o programa irá produzir como saída, a partir do primeiro printf, o qual foi usado como exemplo. Faça o teste de mesa!

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

```
int main() {
```

```
    int i, j, k, y, z;
```

```
    printf("start\n");
```

```
    j = 2;
```

```
    while(j >= 0) {
```

```
        i = j + 2;
```

```
        for(y = 0; y < 2; y++) {
```

```
            k = 0;
```

```
            if(j >= 1 && j < 2) {
```

```
                z = (j + i) * 2;
```

```
                if(k) {
```

```
                    printf("U");
```

```
                } else if(y == 2 || y == 0) {
```

```
                    printf("F");
```

```
                } else {
```

```
                    printf("F");
```

```
                }
```

```
                if(j > 0 && y == 0) {
```

```
                    printf("S");
```

```
                }
```

```
            }
```

```
            else {
```

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

```
            }
```

```
        }
```

```
        j--;
```

```
        printf("\n");
```

```
    }
```

```
    printf("fim");
```

```
    return 0;
```

```
}
```

start

4) (3.0) Faça um programa em linguagem C que receba três números inteiros. Os dois primeiros números representam o começo e o fim de um intervalo, respectivamente e inclusive. O terceiro número, chamado pivô, indica um valor existente dentro do intervalo. O programa imprime um número existente dentro do intervalo que é divisível por 3 e está mais próximo do pivô. Abaixo estão alguns exemplos:

Ex.: Entrada: 1, 9 e 5. Saída: 6

No exemplo acima, o intervalo é entre 1 e 9: 1 2 3 4 [5] 6 7 8 9. O pivô é 5. O número mais próximo de 5 que é divisível por 3 é 6.

Ex.: Entrada: 10, 16 e 13. Saída: 12

No exemplo acima, o intervalo é entre 10 e 16: 10 11 12 [13] 14 15 16. O pivô é 13. O número mais próximo de 13 que é divisível por 3 é 12.