

PUC-Rio – Programação 2 – INF1007
Prova 1 – Soluções – Turma 33E

- Questão 1: (Thales)

```
void contaEleitoresOpcionais(int *idades, int n, int *maiores, int *menores) {
    int i;
    *maiores = 0;
    *menores = 0;
    for (i = 0; i < n; i++) {
        if (idades[i] < 18) (*menores)++;
        if (idades[i] > 65) (*maiores)++;
    }
}

int *obtemEleitoresObrigatorios (int *insc, int *idades, int n, int *obrigatorio) {
    int *v;
    int i, k = 0;
    *obrigatorio = 0;
    for (i = 0; i < n; i++)
        if (idades[i] >= 18 && idades[i] <= 65)
            (*obrigatorio)++;
    if (*obrigatorio == 0) return NULL;
    v = (int *) malloc ((*obrigatorio) * sizeof(int));
    if (v == NULL) {
        printf("Erro na alocação de memória.\n");
        return NULL;
    }
    for (i = 0; i < n; i++) {
        if (idades[i] >= 18 && idades[i] <= 65) {
            v[k] = insc[i];
            k++;
        }
    }
    return v;
}
```

- Questão 2a: (Bruno Reis)

```
Paciente *criaNovoPaciente (char *nome, int idade, char *plano) {
    Paciente *novo;
    novo = (Paciente *) malloc(sizeof (Paciente));
    if (novo == NULL) {
        printf("Memória insuficiente\n");
        return NULL;
    }
    strcpy(novo->nome, nome);
    novo->idade = idade;
    strcpy(novo->plano, plano);
    return novo;
}
```

- Questão 2b: (Gabriel Siqueira)

```
int processaCadastroDosPlanos(PlanoDeSaude *plano, int nplano,
                              Paciente **paciente, int npaciente) {
    int i, j, somaplano = 0;
    for (i = 0; i < npaciente; i++) {
        for (j = 0; j < nplano, j++) {
            if (strcmp(paciente[i]->plano, plano[j].nomeDoPlano) == 0) {
                plano[j].numDePacientes++;
                somaplano++;
            }
        }
    }
    if (somaplano != npaciente)
        return 0;
    return 1;
}
```

- Questão 2c: (Guilherme Berger)

```
int contaPlanosComMaisPacientes(PlanoDeSaude *planos, int numplanos, int limite) {
    if (numplanos == 0)
        return 0;
    return (planos[0].numDePacientes > limite) +
        contaPlanosComMaisPacientes(planos + 1, numplanos - 1, limite);
}
```