

Observações gerais:

- 1) Preencher os dados do cabeçalho do bloco de provas e na folha de questões o nome e a turma;
- 2) Ao final da prova entregar ao professor ou fiscal a folha de questões e o caderno de prova;
- 3) Cada quesito vale cinco pontos.

1º quesito: Uma confederação de desportos deseja identificar os atletas mais bem sucedidos de cada país nos jogos pan-americanos de 2013. Assuma que o atleta mais bem sucedido possui a maior quantidade de medalhas (independentemente do tipo). Existem 2 arquivos disponíveis com os resultados dos jogos:

1. "pais.bin" - Um arquivo binário do tipo Pais, que possui todos os países participantes dos jogos;
2. "atletas.bin" - Um arquivo binário do tipo Atleta, que possui todos os atletas dos países e as respectivas quantidades de medalhas obtidas.

Pais	Atleta	MelhorAtleta
codPais: Número inteiro	nome: String com 50 caracteres	nomePais: String com 30
nome: String com 30 caracteres	medOuro: Inteiro (qtd. de medalhas de ouro)	nome: String com 50 caracteres
sigla: String com 3 caracteres	medPrata: Inteiro (qtd. de medalhas de prata)	numMed: Inteiro que representa a
	medBronze: Inteiro (qtd. de medalhas de bronze)	quantidade de medalhas
	codPais: Inteiro (código do país do atleta)	

A partir da leitura destes dois arquivos crie um arquivo "relatorio.bin" com os atletas mais bem sucedidos de cada país. No novo arquivo devem ser armazenadas informações dos melhores atletas por meio de variáveis do tipo MelhorAtleta, contendo o nome do país, o nome do atleta, e a quantidade total de medalhas obtidas.

Obs.: Os arquivos que já existem serão abertos em modo leitura "rb", e o arquivo que não existe será aberto em modo escrita "wb". Lembre-se que para voltar ao início de um arquivo existe o comando *void rewind(FILE *a)*.

A cópia de cadeia de caracteres é feita com *char* strcpy(char* s1, char* s2)* que copia *s2* em *s1*.

Consideramos que para cada país existe pelo menos um atleta.

2º quesito: Construa um programa em C que realize as seguintes operações:

- a) Faça uma função *int* AlocaMemoria(int n)* que receba um valor *n* como parâmetro e crie dinamicamente um vetor de *n* elementos inteiros e retorne um ponteiro para o início deste vetor;
- b) Crie uma função *void imprime(int *p, int n)* que receba um ponteiro para um vetor e um valor *n* e imprima os *n* elementos desse vetor;
- c) Construa uma função *void LiberaMemoria(int* p)* que receba um ponteiro para um vetor e libere (usando *free()*) esta área de memória;
- d) Crie a função principal que leia um valor *n* do usuário e chame a função *AlocaMemoria*. Depois, a função principal deve ler os *n* elementos desse vetor. Então, a função principal deve chamar a função *Imprime* para impressão dos *n* elementos do vetor criado e, finalmente, liberar a memória alocada através da função *LiberaMemoria*.

3º quesito: Escreva um programa em C que solicite do usuário:

- a) Um valor *n* tal que $1 \leq n < 10$ (assumimos que o usuário vai digitar o valor correto, não é necessária validação);
- b) Os valores para o preenchimento da matriz de números inteiros de tamanho *n x n*;

Depois de preenchida a matriz, o programa deve imprimir na tela:

- c) - A soma dos valores das duas diagonais;
- d) - A soma dos valores de cada linha;
- e) - A soma dos valores de cada coluna;
- f) - A mensagem "Quadrado mágico" caso todas as somas sejam iguais.