

Introdução à Computação

Prof. Roberto Hirata Jr. e Walter Figueiredo Mascarenhas

Segundo Exercício Programa – Data de entrega: 17/12/2004

O objetivo desse exercício programa é calcular e imprimir um calendário formatado. Isto é, dado um ano (um número inteiro entre 1753 - 5000), e um mês (um número de 1 a 12), o programa irá imprimir o calendário formatado com:

- o nome do mês e o ano na primeira linha do cabeçalho,
- os nomes abreviados dos dias da semana (use Do, Se, Te, Qu, Qi, Se, Sa) na segunda linha do cabeçalho,
- os dias do mês desejado formatados de acordo com o dia da semana em que eles caem. As informações devem ser separadas por sinal de tabulação (\t).

Caso não seja dado nenhum mês, o programa deverá imprimir o calendário do ano todo com o mesmo formato acima, mas agora imprimindo 3 meses em cada linha.

Por facilitar a correção, o programa receberá os parâmetros via linha de comando, o ano precedido por “-y” e o mês por “-m”. A saída deverá ser em arquivo, cujo nome é dado como parâmetro, precedido por “-o”.

Por exemplo:

```
> imprimeCalendario -y 2004 -m 12 -o dezembro.txt
```

A saída gerada é um arquivo contendo o seguinte texto:

```
Dezembro 2004
Do Se Te Qu Qi Se Sa
      1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31
```

```
> imprimeCalendario -y 2004 -o ano2004.txt
```

A saída gerada é um arquivo contendo o seguinte texto:

Licenciatura em Matemática - IME - USP

2004

Janeiro							Fevereiro							Março						
Do	Se	Te	Qu	Qi	Se	Sa	Do	Se	Te	Qu	Qi	Se	Sa	Do	Se	Te	Qu	Qi	Se	Sa
				1	2	3	1	2	3	4	5	6	7		1	2	3	4	5	6
4	5	6	7	8	9	10	8	9	10	11	12	13	14	7	8	9	10	11	12	13
11	12	13	14	15	16	17	15	16	17	18	19	20	21	14	15	16	17	18	19	20
18	19	20	21	22	23	24	22	23	24	25	26	27	28	21	22	23	24	25	26	27
25	26	27	28	29	30	31	29							28	29	30	31			

Abril							Maio							Junho						
Do	Se	Te	Qu	Qi	Se	Sa	Do	Se	Te	Qu	Qi	Se	Sa	Do	Se	Te	Qu	Qi	Se	Sa
				1	2	3							1			1	2	3	4	5
4	5	6	7	8	9	10	2	3	4	5	6	7	8	6	7	8	9	10	11	12
11	12	13	14	15	16	17	9	10	11	12	13	14	15	13	14	15	16	17	18	19
18	19	20	21	22	23	24	16	17	18	19	20	21	22	20	21	22	23	24	25	26
25	26	27	28	29	30		23	24	25	26	27	28	29	27	28	29	30			
							30	31												

Julho							Agosto							Setembro						
Do	Se	Te	Qu	Qi	Se	Sa	Do	Se	Te	Qu	Qi	Se	Sa	Do	Se	Te	Qu	Qi	Se	Sa
				1	2	3	1	2	3	4	5	6	7				1	2	3	4
4	5	6	7	8	9	10	8	9	10	11	12	13	14	5	6	7	8	9	10	11
11	12	13	14	15	16	17	15	16	17	18	19	20	21	12	13	14	15	16	17	18
18	19	20	21	22	23	24	22	23	24	25	26	27	28	19	20	21	22	23	24	25
25	26	27	28	29	30	31	29	30	31					26	27	28	29	30		

Outubro							Novembro							Dezembro							
Do	Se	Te	Qu	Qi	Se	Sa	Do	Se	Te	Qu	Qi	Se	Sa	Do	Se	Te	Qu	Qi	Se	Sa	
					1	2			1	2	3	4	5	6				1	2	3	4
3	4	5	6	7	8	9	7	8	9	10	11	12	13	5	6	7	8	9	10	11	
10	11	12	13	14	15	16	14	15	16	17	18	19	20	12	13	14	15	16	17	18	
17	18	19	20	21	22	23	21	22	23	24	25	26	27	19	20	21	22	23	24	25	
24	25	26	27	28	29	30	28	29	30					26	27	28	29	30	31		
31																					

Para facilitar seu trabalho seus professores montaram um arquivo com todos os headers necessários para você fazer o programa. Além disso, parte do programa principal (a parte da interpretação dos parâmetros de entrada) e parte da função de escrita em arquivo está pronta.

E1. Crie uma função `anoBissexto` que recebe como parâmetro um inteiro com um ano válido e retorna 1 se o ano é bissexto, ou 0 caso contrário. Use o seguinte protótipo:

```
int anoBissexto(int ano)
```

- E2.** Crie uma função `diasMes` que recebe como parâmetros dois números inteiros `mes` e `ano` (válidos) e retorna o número de dias do mês `mes` no ano `ano`. Use o seguinte protótipo: `int diasMes(int mes, int ano)`
- E3.** Crie uma função `nomeMes` que recebe como parâmetro um inteiro `mes` com um mês válido e um vetor de caracteres e preenche o vetor com o nome daquele mês, terminado com 0 (isto é, uma string com o nome daquele mês). Use o seguinte protótipo: `void nomeMes(int mes, char nomeMes[12])`
- E4.** Crie uma função `diaSemana` que recebe dois inteiros `mes` e `ano` (mês e ano válidos) e retorna um número de 1 a 7 que indica em que dia da semana cai o primeiro dia do mês `mes` do ano `ano`. Considere que o dia 1 de outubro de 1752 num domingo. Uma maneira mais simples é fazer a contagem a partir de 1 de janeiro de 1800, uma quarta-feira (use o que preferir)¹. Use o seguinte protótipo: `int diaSemana(int mes, int ano)`
- E5.** Complete a função `imprimeCalendario` que recebe como parâmetros um ponteiro para um arquivo e dois inteiros, `mes` e `ano` (mês e ano válidos), chame as funções acima de forma a produzir o resultado desejado. Use o seguinte protótipo: `void imprimeCalendario(FILE *arq, int mes, int ano)`
- E6.** Complete o programa principal `main` para que ele verifique se os argumentos de entrada (`mes`, `ano` e arquivo de saída) são válidos e chame as funções acima para produzir o resultado desejado.

Observações:

1. O programa e as funções devem ser entregues em arquivo único (primeiro as funções e depois o programa principal) devidamente documentados.
2. O cabeçalho do programa deve conter: o seu nome completo, seu número USP, e a turma (47 ou 48). Uma informação por linha, por favor.
3. A indentação correta das funções e do programa será considerada.
4. O exercício é **individual**. Exercícios que indiquem o contrário serão tratados como quebra do código de ética discente da USP e receberão nota 0 (zero).

¹Você sabia que 10 dias do calendário foram “apagados” durante o ajuste do calendário juliano para o calendário gregoriano em outubro de 1582? Nesse ano, o papa Gregório XIII decretou que o dia seguinte ao dia 4 de outubro, que era uma quinta-feira no calendário juliano, seria o dia 15 de outubro, uma sexta-feira. Como mudanças de calendário geram discussões e brigas, o novo calendário só foi adotado na Inglaterra em setembro de 1752. Nesse ano, setembro teve 13 dias a menos e o dia seguinte ao dia 3 (quarta-feira) foi o dia 14 (quinta-feira). A principal diferença entre os calendários é que no calendário juliano, todo ano múltiplo de 4 era bissexto.