

MC102 – Algoritmos e Programação de Computadores

```

      .o.      .oooooooo.  .oooooo.  ooooo  ooooo      .o.      ooooooooo.  ooooooooo
     .ooo.     .ooo'  ooo  .ooo  ooo  .oo  .oo      .ooo.     .oo  ooo.  o .oo  o
    .ooooo.     .oooo.     .ooo      .oo  .oo      .ooooo.     .oo  'ooo.  .oo
   .oo  ooo.     .ooo.     .ooo      .oo  .oo      .oo  ooo.     .oooooooo.  .oo
  .oooooooooooo.  .oooo.     .ooo      .oo  .oo      .oooooooooooo.  .oo.oo.  .oo
 .oo  ooo.  ooo  'ooo.  .ooo  ooo  .oo  .oo      .oo  ooo.  .oo  ooo.  .oo
oooo  oooooo  .oooooooo.  .oooooo.  ooooo  ooooo  ooooo  ooooo  ooooo  ooooo

```

Formas geométricas simples

Nesta tarefa, vamos praticar comandos repetitivos fazendo desenhos com caracteres ASCII. As formas irão variar de acordo com dois parâmetros:

- valor **base** para o número de caracteres e
- caractere utilizado para o desenho.

Triângulo Retângulo: a primeira linha contém 1 caractere, a segunda 3 e assim por diante até atingir o valor base.

Triângulo Retângulo		
base = 3	base = 5	base = 7
.	o	h
...	ooo	hhh
	ooooo	hhhhh
		hhhhhhh

Triângulo Retângulo Invertido: a primeira linha contém o número base caracteres, a segunda base-2 e assim por diante até atingir 1 caractere.

Triângulo Retângulo Invertido		
base = 3	base = 5	base = 7
\$\$\$	&&&&&	0000000
\$	&&&	00000
	&	000
		0

Triângulo Isósceles: o número de caracteres varia da mesma forma que no triângulo retângulo, mas a ponta do triângulo está centralizada com relação à base.

Triângulo Isósceles		
base = 3	base = 5	base = 7

C	W	A
CCC	WWW	AAA
	WWWW	AAAAA
		AAAAAAA

Triângulo Isósceles Invertido: o número de caracteres varia da mesma forma que no triângulo retângulo invertido, mas a ponta do triângulo está centralizada com relação à base.

Triângulo Isósceles Invertido		
base = 3	base = 5	base = 7
mmm	xxxxx	1111111
m	xxx	11111
	x	111
		1

Ampulheta: une um triângulo isósceles invertido a um triângulo isósceles, ambos de mesma base, sendo que os triângulos compartilham a linha com apenas um caractere.

Ampulheta		
base = 3	base = 5	base = 7
iii	!!!!!	8888888
i	!!!	88888
iii	!	888
	!!!	8
	!!!!!	888
		88888
		8888888

Estrela: uma estrela de seis pontas combina um triângulo isósceles, duas ampulhetas e um triângulo isósceles invertido, todos com tamanho base. Observe os exemplos abaixo.

Estrela					
base = 3		base = 5		base = 7	
%		#		*	
% % % % % % % %		###		* * *	
% %		#####		* * * * *	
% % % % % % % %		### ###		* * * * * * * * * * * * * * * * * * * *	
%		# #		* * * * *	* * * * *
		### ###		* * *	* * *
		#####		*	*
		###		* * *	* * *
		#		* * * * *	* * * * *
				* * * * * * * * * * * * * * * * * * * *	
				* * * * *	
				* * *	
				*	

Descrição da entrada:

A entrada para o programa será da seguinte forma:

```
<tipo_do_objeto>  
<base>  
<caractere>
```

As letras que identificarão os tipos dos objetos serão as seguintes:

- **TR**: Triângulo Retângulo
- **TRI**: Triângulo Retângulo Invertido
- **TI**: Triângulo Isósceles
- **III**: Triângulo Isósceles Invertido
- **A**: Ampulheta
- **E**: Estrela

Caso o `tipo_do_objeto` lido não corresponda a um destes, deve ser emitida a mensagem `Objeto inválido`.

O valor para a **base** deve ser um número ímpar e maior ou igual a 3. Caso o número lido não tenha estas características deve ser emitida a mensagem **Base inválida.**

Testes para o SuSy

[illegible]

	*		
arq9.in	TI 2 .	Base inválida.	arq9.res

Esta tarefa inclui mais três testes fechados, que são variações de alguns dos testes já apresentados.

Dicas de Python 3 para esta tarefa:

- Você pode imprimir vários caracteres iguais utilizando comandos como: `print(5 * "*")`
- Você pode imprimir uma cadeia de caracteres sem imprimir uma quebra de linha utilizando: `print("*****", end='')`
- Utilize o gerador de sequências `range(inicio, fim, passo)`. Veja um exemplo em ordem crescente e outro em ordem decrescente.

```
>>> for i in range(1,8,2) :  
    print(i)
```

```
1  
3  
5  
7
```

```
>>> for i in range(7,0,-2) :  
    print(i)
```

```
7  
5  
3  
1
```

- Como na tarefa anterior, a saída pode conter caracteres com acento. Para não ter problemas com a codificação coloque no início do seu arquivo a seguinte linha:

```
# -*- coding: utf-8 -*-
```

Orientações para submissão

Veja [aqui](#) a página de submissão da tarefa. Lembre-se que o arquivo a ser submetido deve se chamar `main.py`. No link [Arquivos auxiliares](#) há um arquivo [args-04.zip](#) que contém todos os arquivos de testes abertos e seus respectivos resultados compactados. Os arquivos `executa-testes.py` e `executa-testes-windows.py` também estão neste pacote.

Todos os alunos matriculados até 29 de março estão inscritos. Se você entrou depois e/ou não estiver cadastrado corretamente, envie email para islene@ic.unicamp.br.

Observe o limite máximo de 20 submissões e que a nota final é proporcional ao número de testes que executaram corretamente.

O peso desta tarefa é 3.

O prazo final para submissão é 22/04/2018. Recomenda-se a realização desta tarefa antes da Prova 1.