



به نام خداوند مهربان

برنامه سازی پیشرفته

دکتر سید ابوالقاسم میر روشندل

پروژه دوم

سفینه

طراح: فاطمه کمانی



سفینه ما که ماموریت اکتشافی اش را با موفقیت انجام داده، قصد دارد تا به سیاره زمین بازگردد و برای این کار باید از میان تعدادی شهاب سنگ عبور کند به او کمک کنید تا نتیجه تصمیماتش را ارزیابی کند.

خواسته :

ما تعداد، مختصات (دو بعدی) و سرعت حرکت شهاب سنگ ها و همچنین مکان های تغییر سرعت سفینه و سرعت پس از آن را به عنوان ورودی به برنامه می دهیم. برنامه ما موظف است در هر واحدی که سفینه طی می کند، زمان و مختصات شهاب سنگ ها را نمایش دهد و سلامت سفینه را چک کند، در صورت برخورد با شهاب سنگ برنامه **متوقف شده** و خبر ناموفق بودن تصمیم را بدهد.

ورودی :

در خط اول به ترتیب تعداد شهاب سنگ ها و در ادامه مختصات هر کدام و سرعت ثابت آنها داده می شود.

در خط دوم ورودی، مکان های تغییر سرعت و سرعت سفینه پس از آن داده می شود.

- توجه داشته باشید که حرکت سفینه در عرض ثابت $y=0$ انجام می شود، اذا در ورودی مکان سفینه، از آوردن این پارامتر تکراری صرف نظر می کنیم. (تنها طول این نقاط ذکر می شود)

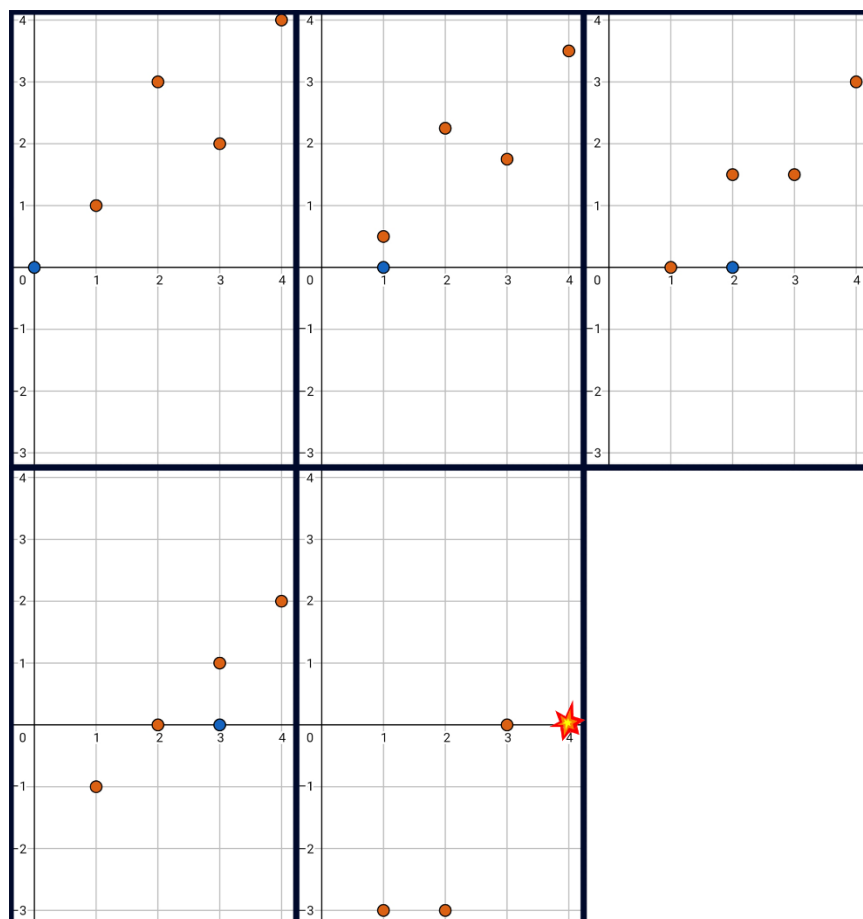
```
Input:~  
4 (1,1)1 (2,3)1.5 (3,2)0.5 (4,4)1~  
0-2_2-1_3-0.5~
```

خروجی :

خروجی باید به ازای هر واحدی که سفینه طی میکند ساخته شود. در ابتدا زمان (t) سپس مختصات سفینه (R) و پس از آن مختصات شهاب سنگ ها (M_i) چاپ شود و وضعیت سلامتی سفینه (safe/crashed) بررسی شود.

Output:-

```
t=0 · R=(0,0) · M1=(1,1) · M2=(2,3) · M3=(3,2) · M4=(4,4) · safe↵  
t=0.5 · R=(1,0) · M1=(1,0.5) · M2=(2,2.25) · M3=(3,1.75) · M4=(4,3.5) · safe↵  
t=1 · R=(2,0) · M1=(1,0) · M2=(2,1.5) · M3=(3,1.5) · M4=(4,3) · safe↵  
t=2 · R=(3,0) · M1=(1,-1) · M2=(2,0) · M3=(3,1) · M4=(4,2) · safe↵  
t=4 · R=(4,0) · M1=(1,-3) · M2=(2,-3) · M3=(3,0) · M4=(4,0) · !CRASHED!↵
```



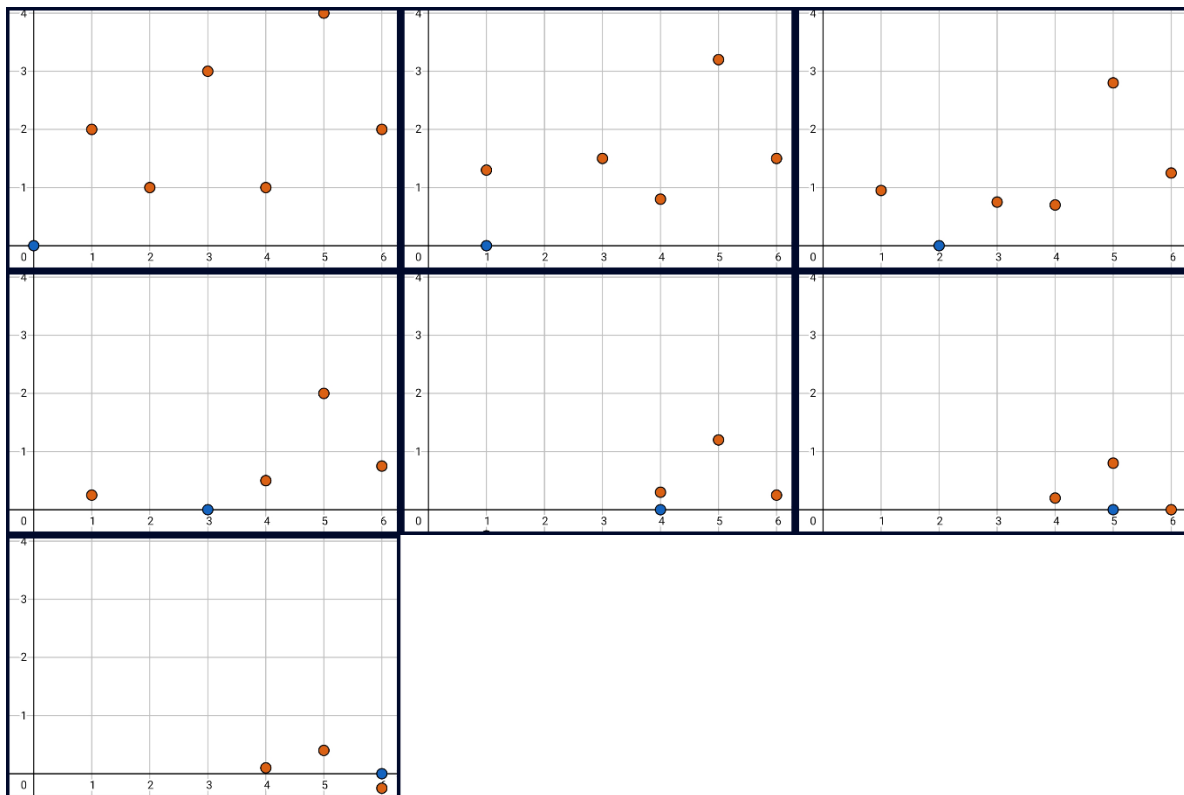
به روایت تصویر ☺

Input: ↵

6 · (1,2) 0.7_ (2,1) 2_ (3,3) 1.5_ (4,1) 0.2_ (5,4) 0.8_ (6,2) 0.5_ ↵
0-1_1-2_2-1_4-2_ ↵

Output: ↵

t=0 · R=(0,0) · M1=(1,2) · M2=(2,1) · M3=(3,3) · M4=(4,1) · M5=(5,4) · M6=(6,2) · safe ↵
t=1 · R=(1,0) · M1=(1,1.3) · M2=(2,-1) · M3=(3,1.5) · M4=(4,0.8) · M5=(5,3.2) · M6=(6,1.5) · safe ↵
t=1.5 · R=(2,0) · M1=(1,0.95) · M2=(2,-2) · M3=(3,0.75) · M4=(4,0.7) · M5=(5,2.8) · M6=(6,1.25) · safe ↵
t=2.5 · R=(3,0) · M1=(1,0.25) · M2=(2,-4) · M3=(3,-0.75) · M4=(4,0.5) · M5=(5,2) · M6=(6,0.75) · safe ↵
t=3.5 · R=(4,0) · M1=(1,-0.45) · M2=(2,-6) · M3=(3,-2.25) · M4=(4,0.3) · M5=(5,1.2) · M6=(6,0.25) · safe ↵
t=4 · R=(5,0) · M1=(1,-0.8) · M2=(2,-7) · M3=(3,-3) · M4=(4,0.2) · M5=(5,0.8) · M6=(6,0) · safe ↵
t=4.5 · R=(6,0) · M1=(1,-1.15) · M2=(2,-8) · M3=(3,-3.75) · M4=(4,0.1) · M5=(5,0.4) · M6=(6,-0.25) · safe ↵



به روایت تصویر ☺

نکات تکمیلی:

- استفاده از توابع دیفالتِ مربوط به استرینگ به جز `length` و `charAt` در این پروژه غیر مجاز می باشد.
- ورودی باید نسبت به فاصله (space) بی تاثیر بماند.
- فرمت ورودی و خروجی باید به صورت گفته شده باشد.
- خروجی باید تا دورقم اعشار چاپ شود.
- نمایش نمودار ها مانند نمونه های بالا در آخر برنامه (به صورت کنسولی) **نمره مثبت** خواهد داشت.

نحوه ارسال پروژه:

- هیچ توجیهی در استفاده از زبان های برنامه نویسی دیگر وجود نداشته، برابر با تحویل ندادن پروژه بوده و **نمره صفر** به دانشجو تعلق می گیرد.
- پروژه تنها به صورت انفرادی قابل انجام بوده و در صورت وجود هرگونه تشابه بین دو کد یا عدم تسلط به روند برنامه در زمان تحویل، **نمره منفی** به دانشجویان داده می شود.
- برای تسلط بهتر و کسب نمره خوب، در این درس به انجام پروژه های اضافه بر پروژه های ترم اقدام کنید.
- زودتر از پایان مهلت پروژه اقدام به ارسال آن نموده و تأیید گرفته شده را بلافاصله چک کنید.
- رعایت فاصله کد از سر خط و همچنین کامنت نویسی مناسب الزامی است.