

(درسنامه) نصب پایتون

به نام خدا، سلام!

به اولین درسنامه خوش اومدید!

با اینکه سر کلاس نحوه‌ی نصب پایتون رو توضیح دادم ولی اگه می‌خواید الان می‌خواید نصبش کنید این بخش به دردتون می‌خوره.

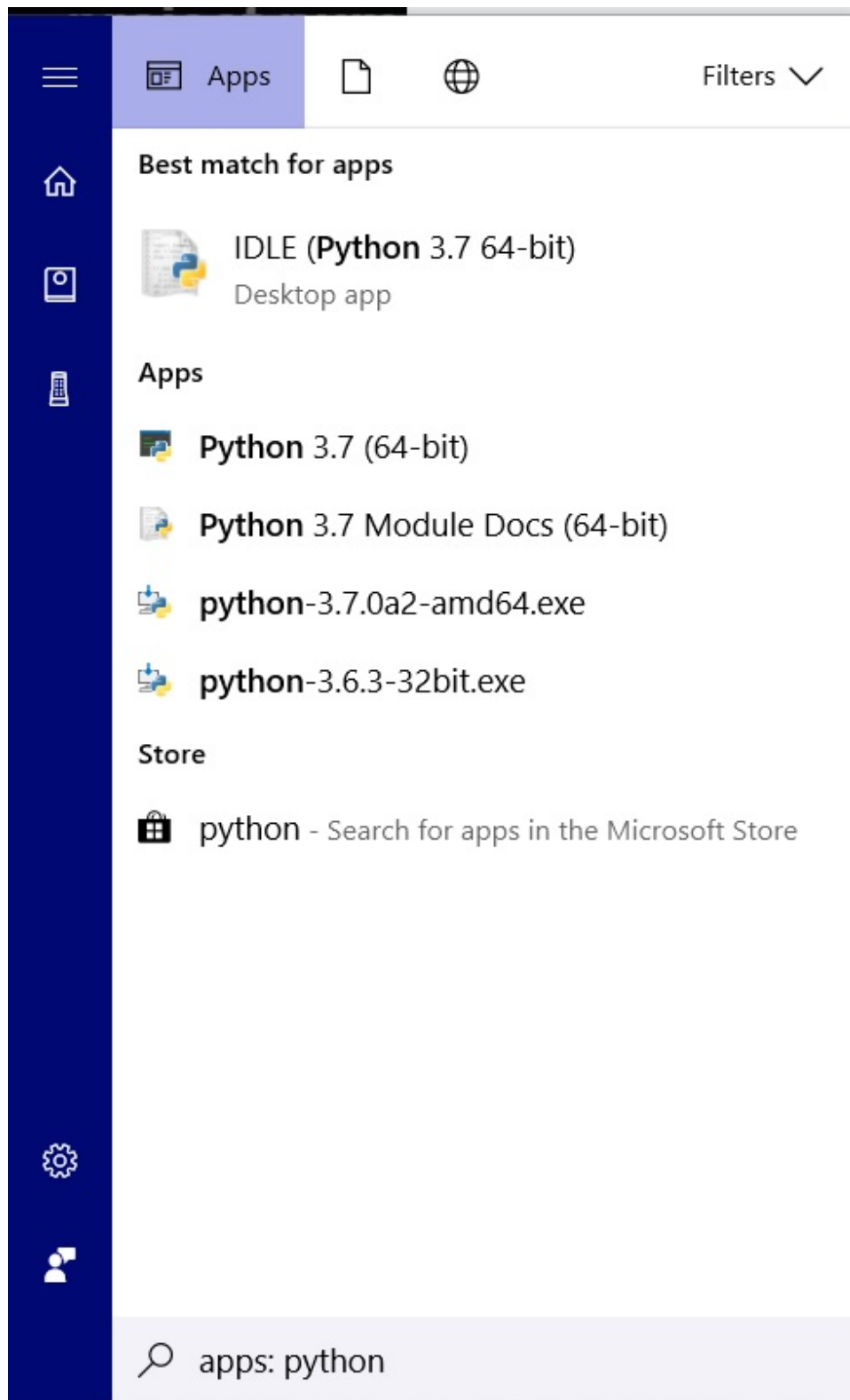
برای اینکه یه کم انگلیسیتون قوی بشه و یادبگیرید که چجوری باید در دنیای برنامه‌نویسی نفس کشید، روی لینک زیر کلیک کنید و شروع کنید راه خودتون رو برای پیدا کردن راه حل مشکلاتون باز کردن!!!

اولین قدم فهمیدن، پرسیدنه! اونم از آقای گوگل.

بعدش هم خودتون برید سراغش :)

توضیحات تکمیلی

وقتی توی استارت پایتون رو سرچ می‌کنید ۲ تا گزینه وجود داره:



Python3 interpreter

مترجم پایتون، برنامه‌ای هست که خط به خط دستورات پایتون رو از ورودی می‌گیره و اجراش می‌کنه.

IDLE

این برنامه در واقع یک IDE یا محیط توسعه‌ی نرم‌افزار هست که می‌تونیم توش برنامه‌مون رو بنویسیم و اون هم به پایتون بگه که اجراش کنه. این برنامه همراه با مترجم پایتون روی سیستم شما نصب شده. در نگاه اول شاید خیلی تفاوتی با مترجم پایتون نداشته باشه ولی چندتا قابلیت و ابزار مثل عوض کردن رنگ نوشته(با توجه به سینتکس)، پیشنهاد ادامه‌ی دستور و ... رو داره. شاید ابتدایی‌ترین قابلیتی که به دردتون بخوره ایجاد یک فایل جدید برای نوشتن برنامه توش هست که از طریق `File>new file` بهش می‌تونید دسترسی داشته باشید.

(درسنامه) کامپیوتر چجوری کار می‌کنه؟

ما تو این کلاس می‌خوایم یاد بگیریم که دقیقا چجوری یک مسئله رو حل کنیم؟ وقتی اینو فهمیدیم می‌تونیم به کامپیوتر هم بگیم که چجوری مسئله رو حل کنه.

الگوریتم

الگوریتم توضیح حل مسئله به صورت مرحله مرحله است. به مثال زیر که یه الگوریتم برای پختن تخم مرغ زده شده است توجه کنید:

Aspects of an Algorithm

- Example: Making scrambled eggs

1. Beating the eggs for 20 to 35 seconds in a bowl
2. Heating a frying pan over a medium-low heat
3. Melt some butter in the frying pan
4. Cook eggs on the pan and stir eggs while cooking
5. Add other ingredients
6. Serve the scrambled eggs

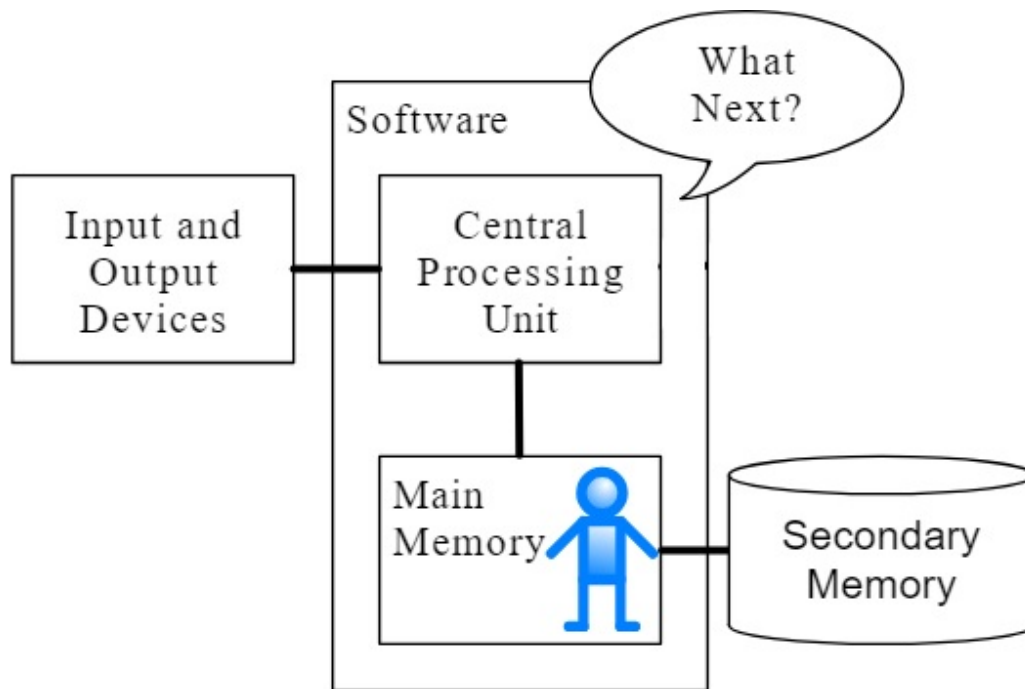


Module 1 : Computing and Algorithms

19 of 39

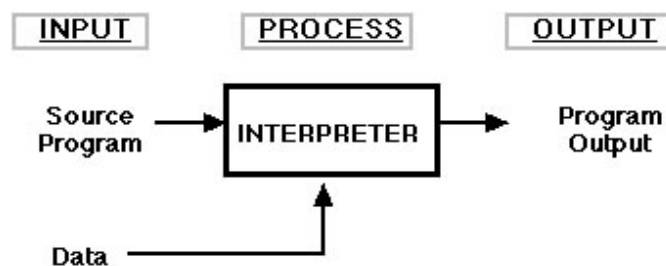
ما باید سعی کنیم که مسئله بزرگ خودمون رو به مسئله‌های کوچیک تر بشکونیم. مسئله‌هایی که حل کردنشون رو بلدیم و می‌تونیم به کامپیوتر بگیم که اینجوری جلو برو. کامپیوتر الگوریتم رو می‌فهمه. برنامه در واقع الگوریتمیه که به زبان کامپیوتر نوشته شده.

شمای کامپیوتر



سی پی یو همواره در حال پرسیدن این سواله که الان باید چیکار کنم. دستورات میرن تو سی پی یو و اجرا می‌شن. سی پی یو اطلاعاتی رو که می‌خواد از روی رم می‌خونه و می‌نویسه. رم خیلی سریعه ولی مشکلش اینکه وقتی برق قطع بشه یعنی کامپیوتر خاموش بشه اطلاعات روش هم می‌پره. برای همین از هارد استفاده می‌کنن که اطلاعات روش ماندگارن ولی کندن. ورودی و خروجی هم اطلاعات تکمیلی برای حل یک مسئله رو به سی پی یو میده. مثل موس و کیبورد و صفحه نمایش و پرینتر و ... مثلاً اگه قراره چیزی که توسط کاربر تایپ می‌شه روی صفحه چاپ شه سی پی یو از طریق IO می‌فهمه که چی تایپ شده.

برنامه!



برنامه‌نویس توسط زبان‌های برنامه‌نویسی سطح بالا کارهایی که کامپیوتر باید به ترتیب انجام بده رو توصیف می‌کنه و کامپایلر یا مترجم اون رو به زبان ماشین در میاره تا سی پی یو بفهمه. مثلاً برنامه نویسی میگه «سلام رو چاپ کن» و کامپایلر اونو تبدیل می‌کنه به کلی دستور ریزتر که سلام بتونه روی صفحه‌ی نمایش چاپ شه. مثل اینکه کجا باید نوشته بشه یا فونتش چجوری باشه یا کدوم پیکسل‌ها باید مشکی بشن ... برنامه‌ی ما درواقع روی رم قرار می‌گیره و هربار سی پی یو از رم می‌خواد که دستور بعدی رو بهش بده تا اجراش کنه.

(درسنامه) مقادیر و عملگرها و گزاره‌ها

نکته ۱: اگر در کدمان از کارکتر # استفاده کنیم، تمامی نوشته‌های بعد از آن تا پایان خط توسط پایتون نادیده گرفته می‌شود. به نوشته‌های بعد از کارکتر مربع اصطلاحاً کامنت گفته می‌شود که همان طور که پیداست برای توضیح دستورات نوشته شده استفاده می‌شود. در ادامه نمونه‌هایی از آن را می‌بینید. نکته ۲: در حین خواندن درسنامه چندتایی تمرین سر کلاس می‌بینید که سر کلاس درمورشون بحث کردیم و حلشون کردیم. جواب‌هاشون رو نمی‌ذارم تا اگه دوباره بهش رسیدید و جوابش رو نمی‌دونستید بگید که جلسه‌ی بعد در مورد بحث کنیم.

مقدار

مقدار یا value خیلی نیاز به توضیح ندارد.

مثلاً ۳ که از نوع عدد صحیح یا integer هست که تو پایتون به اختصار int بهش می‌گن.

یا کلمه‌ای مثل «salam» که یه رشته از کارکترهاست و تو پایتون بهش می‌گن string

برای فهمیدن نوع یه چیز می‌تونیم از دستور type استفاده کنیم

```
>>> 3
3
>>> 5+6
11
>>> 3
3
>>> 5+6
11
>>> 6*7
42
>>> 8/4
2.0
>>> 8/3
2.6666666666666665
>>> 10-100
-90
>>> 2**8 # tavan
256
>>> 17//5 # taghsime sahih
3
>>> 17%5 # baghimande
2
```

```

>>> 'hello'
'hello'
>>> "aleikom" # " mitoone bejaye ' estefade beshe
'aleikom'
>>> type(4)
<class 'int'>
>>> type(-1)
<class 'int'>
>>> type("aleikom")
<class 'str'>
>>> type('hello')
<class 'str'>

```

تمرین سر کلاس

نوع 7/5؟

نوع 8/2؟

نوع 2//8؟

نوع 4.5؟

عملگرها

عملگرها داری اولویت ریاضی هستند که به ترتیب زیر می باشد:

```

()
\**
\* /
+ -

```

برای مثال توجه کنید

```

>>> 2 ** 2 * 3 # = 4 * 3
12
>>> 2 * 2 ** 3 # = 2 * 8
16
>>> 2 * 3 + 1 # = 6 + 2
7
>>> 2 * ( 3 + 1 ) # = 2 * 4
8

```

تمرین سر کلاس

حاصل مقادیر زیر؟

```
( 1 + 2 ) / 3
( 1 + 2 ) // 3
5 ** (3 + 2)
(3 + 2) ** 2 / 5
```

عملگرهای رشته

همچنین اپراتورهای +,* برای استرینگ هم پشتیبانی می‌شود

به علاوه در واقع همان نقش چسباندن و ضرب نقض تکرار را دارد:

```
>>> 'salam ' + 'aleykom'
'salam aleykom'
>>> 'tekrar' * 3
'tekrartekrartekrar'
```

تمرین سر کلاس

```
"very" * 3 + "simple"
```

متغیرها

متغیر یا variable به خونه‌ای از حافظه می‌گن که ما تو برنامه‌مون ازش استفاده می‌کنیم. می‌تونیم به این خونه از حافظه یه اسم اختصاص بدیم که باید این ۳ تا قانون رو رعایت کنن: ۱. تنها می‌تواند از اعداد، حروف انگلیسی و _ استفاده کند. ۲. با عدد شروع نشود. ۳. مانند کلمات زیر که به آن‌ها keyword یا کلمات رزرو شده‌ی پایتون (که همان دستورات اصلی این زبان هستند) نباشند:

| | | | | |
|----------|---------|--------|----------|-------|
| and | del | from | None | True |
| as | elif | global | nonlocal | try |
| assert | else | if | not | while |
| break | except | import | or | with |
| class | False | in | pass | yield |
| continue | finally | is | raise | |
| def | for | lambda | return | |

متغیرها به درد بخور تریت مفهوم توی برنامه نویسی هستن چون میشه به بهشون مقداری رو نسبت داد و در ادامه ازشون استفاده کرد.

برای مثال به دستورات زیر نگاه کنید:

```
>>> x = 1 # assignment: yani 1 ro beriz too motaghayeri be name x
>>> x
>>> x = x + 1 # aval samte rast hesab shode va meghdare an dar chap
rikhte mishavad
>>> x
2
```

```
>>> hello = 'salam'
>>> type(hello)
<class 'str'>
```

```
>>> gheymate_ivan = 100
>>> tedad = 20
>>> gheymate_kol = gheymate_ivan * tedad
>>> gheymate_kol
2000
>>> tedad
20
```

نکته : سعی کنید اسم متغیرها تون واضح و توضیح دهنده ی کارکردش باشه و اگه چند بخشیه از _ استفاده کنید

###تمرین سر کلاس

خروجی برنامه های زیر چیست؟

```
day = 10
month = 'mordad'
day + month
day = '10'
day + month
```

```
afarin = "afarin"
sadhezar_afarin = afarin * 100000
sadhezar_afarin
```

کدوم یک از متغیرهای زیر قابل قبولند؟

xd011
xd0l
salam_22
yield
kelase_python
kelasePython

متغیر

0.3f

مقادیر ۲ متغیر را با هم عوض کنید.

راهنمایی: از متغیر کمکی استفاده کنید.

(درسنامه) ورودی گرفتن و خروجی دادن

در دنیای واقعی ما می‌خوایم برنامه‌هایی بنویسیم که نسبت به شرایط واکنش‌های متفاوت نشون بدن نه اینکه همیشه به مقدار ثابت رو نمایش بدن یا به کار ثابت رو انجام بدن.

ورودی

یکی از راه‌های ورودی گرفتن، گرفتن چیزی هست که کاربر اونو تایپ می‌کنه در پایتون دستوری وجود دارد به نام `input()` که این دستور از کاربر رشته‌ای رو ورودی می‌گیره و استرینگ آنرا خروجی می‌ده

```
>>> voroodi = input()
yek chizaii
>>> voroodi
'yek chizaii'
```

مثال: برنامه ای بنویسید که اسم کاربر را ورودی گرفته و به او سلام کند!

```
>>> name = input()
>>> "salam "+name
```

همچنین می‌توان یک رشته را به عنوان ورودی دستور داد و این رشته قبل از گرفتن ورودی نمایش داده می‌شه.

```
>>> input("enter something ")
enter something ....
```

خروجی

همچنین برای نمایش یک متن از دستور `print` استفاده می‌شود

```
>>> print("hi")
hi
>>> print(3)
3
>>> a = "reshte"
>>> print(a)
reshte
```

حال برنامه‌ی مثال رو بهتر می‌نویسیم:

```
1 | name = input("what is your name? ")
2 | print("hi "+name)
```

ورودی که input به ما می‌دهد یک رشته است. برای تبدیل آن به عدد می‌توان از دستور int استفاده کرد

```
age = input("what is you age? ")
age_int = int(age)
next_age = age_int + 1
print("your age next year will be: ")
print(next_age)
```

همینو همیشه بهتر نوشت:

```
age = input("what is you age? ")
age = int(age)
print("your age next year will be: ")
print(age+1)
```

تمرین سر کلاس

چه چیزی چاپ می‌شود؟

```
x = 5
x = x + 1
x = 5 * 2
x * 3
print(x)
```

چه چیزی چاپ می‌شود؟

```
hazine_zamin_metri = 10
masahate_zamin = 20
hazineye_zamin = hazine_zamin_metri * masahate_zamin
masahate_zamin = 30
print(hazineye_zamin)
```

برنامه‌ای بنویسید که شعاع پیتزا را بخواند و مساحت آن را نمایش دهد.

بیک ۲ عدد بخواند و مقادیر زیر را محاسبه کند $n^m/2$ و $((n-2)/3)^m$

بیک یک عدد از ۰ تا ۳۱ بخواند و عدد ۵ رقمی معادل در مبنای دوی آن را چاپ کند.

بیک یک عدد دو رقمی بخواند و برعکسش را چاپ کند؟

☆ چندگزینه‌ای ها

پاسخ پرسش‌های چند گزینه‌ی زیر رو به هر فرمتی که دوست دارید به صورت یک فایل پی دی اف آپلود کنید!

راهنمایی: در استفاده از پایتون آزادید!

۱. کدام اسم برای متغیر مناسب است؟ (چند جواب دارد)

راهنمایی: برای تست می‌تونید در محیط برنامه نویسی پایتون به متغیر خود یک مقدار دهید و ببینید که آیا با خطا مواجه می‌شوید یا نه؟

1. firstName
2. first_name
3. _first_name
4. _
5. __
6. 0isZero
7. zeroIs0
8. in

۲. متغیر `zero = 0` با کدام متغیر برابرست؟

1. `zero + 0`
2. `Zero`
3. `_zero`
4. `0`
5. `zero + '0'`

۳. مقدار زیر برابر با کدام است؟

`3*2**4*2`

۱. 2592

۲. 1679616

۳. 96

۴. 768

☆☆ سوال زرد

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه (حداکثر میزان استفاده از رم): ۲۵۶ مگابایت

نکته: برای اینکه کار با کوئرا رو یاد بگیرید این سری آموزش رو حتما ببینید!

مهدی که از کدزدن خسته شده است، به تازگی به رشته‌ی صنایع علاقه پیدا کرده است. به همین دلیل تصمیم گرفته است تا در مورد این رشته تحقیق کند. او به افراد مختلفی مراجعه می‌کند و هرکدام یک مقداری اطلاعات به او می‌دهند. او به اندازه‌ی مقدار اطلاعاتی که از اشخاص می‌گیرد متعجب می‌شود. مثلاً اگر یک عدد اطلاعات بگیرد می‌گوید Wow! ، اگر دوتا اطلاعات بگیرد می‌گوید Woow! و به همین شکل مقدار کشیدن کلمه (تعداد o ها) زیاد می‌شود. حالا شما باید بگویید که اگر یک نفر به اندازه‌ی n به مهدی اطلاعات بدهد، ما باید انتظار چه کلمه‌ای را از او داشته باشیم.



ورودی

در تنها سطر ورودی یک عدد طبیعی n به شما داده شده است که نمایانگر مقدار اطلاعات داده شده به مهدی است.

$$1 \leq n \leq 10$$

خروجی

خروجی شامل یک کلمه است، که نشان‌دهنده‌ی کلمه ایست که مهدی بعد از شنیدن اطلاعات راجع به رشته‌ی صنایع می‌گوید.

مثال

ورودی نمونه ۱

1

خروجی نمونه ۱

Wow!

ورودی نمونه ۲

2

خروجی نمونه ۲

Woow!

راهنمایی

برای نوشتن یک برنامه باید الگوریتم آن را پیدا کنید. و پیدا کردن الگوریتم یعنی هنر تشخیص الگوهای تکراری و جمع و جور کردن آن‌ها!

☆ برعکس کردن سه رقمی

- محدودیت زمان: ۱.۵ ثانیه
- محدودیت حافظه: ۱۰۰ مگابایت

در کلاس نحوه‌ی برعکس کردن یک عدد ۲ رقمی رو دیدیم. حال می‌خواهیم که یک عدد ۳ رقمی را وارونه کنیم.

ورودی

ورودی تنها شامل یک خط است که در آن یک عدد طبیعی n حداکثر سه رقمی آمده است.

$$1 \leq n, \leq 999$$

خروجی

خروجی برنامه‌ی شما باید شامل ۱ خط باشد که مقدار صحیح وارون عدد ورودی باشد.

مثال

ورودی نمونه ۱

123

خروجی نمونه ۱

321

ورودی نمونه ۲

53

خروجی نمونه ۲

350

۵۳ معادل ۰۵۳ می‌باشد که وارون آن ۳۵۰ می‌شود.

☆☆ درهم‌سازی حسنکی

راهنمایی: نحوه‌ی شکستن به مسائل کوچک تر

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: 100 مگابایت

حسنگ تازه یاد گرفته است که برنامه‌ی پایتون بنویسد. به دلیل مطالعات گسترده و خلاقیت بی‌نهایت او، برای خود یک روش درهم‌سازی ابداع کرده است که بسیار سریع کار می‌کند و به همان اندازه به درد نخور است!

برنامه ای بنویسید که یک عدد بگیرد و یک عدد سه رقمی نمایش دهد به روش درهم‌سازی حسنکی تولید شده باشد. این روش درهم‌سازی این صورت است که یکان عدد پاسخ برابر با قیمانده‌ی عدد ورودی به ۲، دهگان باقیمانده به ۳ و صدگان باقیمانده به ۴ باشد.

درسنامه

همون طور که رشته به عدد صحیح اینگونه تبدیل میشد

```
>>> int(string_value)
```

مقدار صحیح هم اینگونه به رشته تبدیل می‌شود:

```
>>> str(integer_value)
```

ورودی

ورودی تنها شامل یک خط است که در آن ۱ عدد طبیعی n آمده است.

$$1 \leq n \leq 1000$$

خروجی

خروجی برنامه‌ی شما باید شامل ۱ خط باشد که عدد خروجی روش درهم‌سازی حسنگ باشد.

مثال

ورودی نمونه ۱

13

خروجی نمونه ۱

111

باقیمانده‌ی ۱۳ به ۲ برابر ۱، به ۳ برابر ۱ و به ۴ برابر ۱ است.

ورودی نمونه ۲

170

باقیمانده‌ی ۱۷۰ به ۲ برابر ۰، به ۳ برابر ۲ و به ۴ برابر ۲ است.

خروجی نمونه ۲

220

#راهنمایی سعی کنید برنامه را به اجزای کوچک‌تری بشکنید. برای اینکار می‌توانید یک بار خودتان برای هر ورودی سعی کنید پاسخ را محاسبه کنید و به فرآیند پاسخ به سوال توجه کنید.

برای مثال اجزای این برنامه عبارت‌اند از:

- خواندن عدد
- محاسبه باقی‌مانده‌ها
- ساخت عدد با باقی‌مانده‌ها
- خروجی دادن پاسخ

☆☆ اولویت عملگرها

پرسش

حسنک پس از رسیدن به خانه از کلاس پایتون سوالی برایش پیش آمد. او اولویت بعضی از عملگرها را نمی‌دانست. به حسنک کمک کنید و بگویید عملگرهای زیر به ترتیب اولویت چگونه می‌باشند؟

/
//
+
%

راهنمایی

میتوانید با طراحی تست‌هایی مانند زیر ۲ به ۲ عملگرها را اولویت بندی کنید:

8 * 4 // 2
8 // 4 * 2

برای مثال از پاسخ‌های یکسان فهمیده می‌شود که عملگر ضرب و تقسیم صحیح در اولویت یکسان هستند.

نحوه پاسخ

تست‌های خود همراه با پاسخ نهایی به صورت یک فایل پی‌دی‌اف یا پایتون بارگذاری کنید. استاد به صورت دستی پاسخ‌های شما را بررسی خواهد کرد.

راهنمایی‌تر (!)

از آقای گوگل هم می‌توانید بپرسید ولی بخشی از نمره مربوط به طراحی تست‌هایی می‌شود که شما کرده‌اید. پس هوشمندانه عمل کنید.

☆☆☆ هب هاب ساده

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۱۲۸ مگابایت

در این مسئله شما باید اعداد ۱ تا ۱۰۰ به طرز مخصوصی چاپ کنید. به این صورت که عدد شما اگر مضرب ۲ بود نشان دهد ۲، اگر مضرب ۳ بود نشان دهد ۳ و اگر مضرب جفتشان بود ۶ در غیر اینصورت ۰ چاپ کند.

ورودی

ندارد.

خروجی

پاسخ شما باید در یک خط چاپ شود.

مثال

برای مثال ۱۰ عدد اول اینگونه باید چاپ شود.

خروجی نمونه

0232060232

راهنمایی

دست به قلم شوید و ببینید آیا الگویی تکراری می‌توانید به دست آورید؟

☆☆☆ جابه‌جایی

پرسش

حسنک پس از ور رفتن با پایتون به کارکرد عجیبی رسید. او برای جابه‌جایی ۲ متغیر بدون متغیر کمکی از روش زیر استفاده کرد.

$$a, b = b, a$$

او پس از کف کردن‌های بسیار و سرچ‌های بسیارتر متوجه شد که روش دیگری نیز برای جابه‌جایی ۲ متغیر از نوع صحیح روش جالب‌تری نیز وجود دارد. آیا می‌توانید آن را حدس بزنید؟

نحوه پاسخ‌دهی

ابتدا ۲ متغیر a و b را مقداردهی کنید. روش خود را پیاده‌سازی کنید. انتظار می‌رود در پایان برنامه مقادیر این ۲ متغیر با هم جابه‌جا شده باشند.

برنامه‌ی پایتون خود را آپلود کنید. به صورت دستی تصحیح صورت می‌پذیرد!

راهنمایی

این سوال بیشتر از یک سوال پایتونی، یک سوال ریاضی است. روش پایتون‌وار (!) آن را دیدید. حال روش ریاضی‌وار آن را پیدا کنید.