

# ساختار و زبان کامپیوتر

استاد: دکتر اکبرزاده  
هد تی ای: خانم غیبی

## پروژه IBM

نیم سال دوم ۱۴۰۲-۱۴۰۳

## 1- حدس کولاتز (ساده)

با ورودی گرفتن عدد  $n$  در هر مرحله، اگر  $n$  زوج بود آن را برابر  $n/2$  و در غیر این صورت برابر  $3n+1$  قرار دهید. دنباله اعداد طی شده تا وقتی عدد مورد نظر 1 شود را خروجی دهید.

### ورودی:

در تنها خط ورودی، عدد  $n$  وارد شده است.

$$n \geq 1$$

### خروجی:

در خروجی باید دنباله اعدادی که طی شده است تا زمانی که عدد فعلی 1 شود، خروجی داده شود.

#### ورودی نمونه 1:

5

#### خروجی نمونه 1:

5 16 8 4 2 1

#### ورودی نمونه 2:

6

#### خروجی نمونه 2:

6 3 10 5 16 8 4 2 1

#### ورودی نمونه 3:

2

#### خروجی نمونه 3:

2 1

## 2- عملیات منطقی (متوسط)

در این سوال، یک رشته در ورودی داده می‌شود که شامل تعدادی عملگر منطقی است. در حل این سوال اولویت عملگرها را در نظر نگرفته و از چپ به راست به ترتیب عملیات‌ها را انجام دهید.

### ورودی:

در یک خط، یک رشته شامل عملگرهای منطقی و تعدادی عدد صحیح مبنای 10 داده می‌شود. در ضمن تضمین می‌شود که اعداد ورودی در ثبات‌های پردازنده IBM جا می‌شوند.

عملگرهای منطقی مورد استفاده در این سوال عبارتند از:

AND: &

OR: |

XOR: ^

NOT: ~

### خروجی:

در یک خط حاصل عملیات منطقی را در مبنای 10 نمایش دهید.

### ورودی نمونه 1:

5|2^7~&2

### خروجی نمونه 1:

2

### توضیحات نمونه 1:

مراحل محاسبه به این صورت است که در ابتدا عدد 5 را با 2 OR کرده و سپس حاصل را با عدد 7 XOR کرده و حاصل کنونی را NOT کرده و در نهایت خروجی را با عدد 2 AND کرده‌ایم.

### ورودی نمونه 2:

-12^34&56|78~

### خروجی نمونه 2:

-95

### 3- برج‌های هانوی (متوسط)

این مسئله شامل سه ستون و تعدادی دیسک مختلف است که می‌توانند بر روی هم در هر ستون قرار بگیرند. پازل با دیسک‌ها به شکل مرتب و در حالت صعودی از نظر اندازه بر روی یکی از ستون‌ها شروع می‌شود، که کوچکترین دیسک، بالای ستون است و یک شکل مخروطی ایجاد می‌کند. یک تابع بنویسید که با داشتن تعداد دیسک‌ها و نام‌های ستون‌ها، مراحل مورد نیاز برای انتقال تمامی دیسک‌ها از ستون اول به ستون آخر را با رعایت قوانین زیر، چاپ کند:

1. هر بار تنها یک دیسک می‌تواند حرکت کند.
2. هر حرکت شامل برداشتن دیسک بالایی از یکی از ستون‌ها و قرار دادن آن بر روی یک ستون دیگر است.
3. هیچ دیسکی نمی‌تواند بر روی یک دیسک کوچک‌تر قرار گیرد.

#### ورودی:

در خط ورودی، عدد  $m$  که نشان دهنده تعداد دیسک‌ها است، وارد شده است.  
در حالت اولیه، تمام  $m$  دیسک در ستون 0 قرار گرفته‌اند.

#### خروجی:

در خروجی باید مراحل که طی شده تا تمامی دیسک‌ها به ستون 2 برسند، چاپ شود و در هر خط دوتایی  $(i,j)$  چاپ شود که یعنی در آن، حرکت بالاترین دیسک ستون  $i$ ام بر روی ستون  $j$  قرار گرفته است. دقت کنید که حرکات، باید امکان پذیر باشند.

#### ورودی نمونه 1:

1

#### خروجی نمونه 1:

(0,2)

#### ورودی نمونه 2:

2

#### خروجی نمونه 2:

(0,1)

(0,2)

(1,2)

#### ورودی نمونه 3:

3

خروجی نمونه 3:

(0,2)

(0,1)

(2,1)

(0,2)

(1,0)

(1,2)

(0,2)

## 4- محاسبه زیرمجموعه‌های یک مجموعه (متوسط)

با ورودی گرفتن عدد  $n$ ، زیرمجموعه‌های مجموعه اعداد 1 تا  $n$  را خروجی دهید.

**ورودی:**

در تنها خط ورودی، عدد  $n$  وارد شده است.

$$n \geq 0$$

**خروجی:**

در خروجی باید در هر خط یک زیرمجموعه از مجموعه اعداد 1 تا  $n$  خروجی داده شود، دقت کنید که ترتیب خروجی دادن شما باید مشابه تست‌های نمونه باشد تا نمره سوال را بگیرید.

**ورودی نمونه 1:**

0

**خروجی نمونه 1:**

{}

**ورودی نمونه 2:**

1

**خروجی نمونه 2:**

{1}

{}

**ورودی نمونه 3:**

3

**خروجی نمونه 3:**

{1, 2, 3}

{1, 2}

{1, 3}

{1}

{2, 3}

{2}

{3}

{}

## 5- محاسبه سطر نام از مثلث خیام پاسکال به صورت بازگشتی (سخت)

در این سوال باید سطر نام مثلث خیام پاسکال را محاسبه و خروجی دهید. برای این کار نیاز است که امکان جمع اعداد بزرگ فراهم باشد (ساخت یک ماشین حساب جمع کننده اعداد بزرگ). برای محاسبه باید از اتحاد پاسکال استفاده کنید:

$$C(n, r) = C(n - 1, r - 1) + C(n - 1, r)$$

### ورودی:

در تنها خط ورودی، عدد  $n$  داده می شود که نشان دهنده index سطری از مثلث خیام پاسکال است که باید خروجی داده شود. (دقت شود که سطر اول مثلث خیام پاسکال، در اصل با  $n = 0$  نمایش داده می شود)

### خروجی:

در تنها خط خروجی، اعداد سطر مدنظر را به ترتیب و با یک فاصله میان اعداد آن خروجی دهید.

### ورودی نمونه 1:

0

### خروجی نمونه 1:

1

### ورودی نمونه 2:

1

### خروجی نمونه 2:

1 1

### ورودی نمونه 3:

5

### خروجی نمونه 3:

1 5 10 10 5 1

## 6- الگوریتم جستجوی اول عمق (DFS) (چالشی - امتیازی)

یک گراف بدون جهت داریم. از راس با شماره 1، جستجوی اول عمق زده و مسیر پیمایش شده در DFS را خروجی دهید. دقت شود که در هر گام که چندین همسایه به عنوان انتخاب برای راس بعدی وجود داشت، به راس دیده نشده با کمترین مقدار برود.

### ورودی :

در خط اول ورودی دو عدد  $n, m$  داده می شود که به ترتیب برابر تعداد رئوس و تعداد یالهای گراف است. در  $m$  خط بعدی در هر خط دو عدد داده می شود که نشان دهنده وجود یال میان دو راس متناظر با آن دو عدد است.

$$n, m \leq 20$$

### خروجی :

با توجه به فرض گفته شده در صورت سوال، مسیر پیمایش شده در DFS را خروجی دهید.

### ورودی نمونه:

7 10

1 5

1 7

3 2

2 4

4 5

6 4

6 7

3 6

3 7

5 6

### خروجی نمونه:

1 5 4 2 3 6 7

### تصویر گراف نمونه:

