احمدرضا ذبیحی به نام خدا ش.د: ۴۰۰۵۲۱۳۴۲

**4.1:**

عدم استفاده از else در برنامه میتواند عواقب متعددی داشته باشد از جمله پوشش ندادن تمام حالات ، باز ماندن شرط و ... اما در این مورد بخصوص چون کل مسعله 6 حالت دارد و تمام این حالات با if و elif پوشش داده شده پس مشکلی پیش نمی اید.

**4.2:**

این موضوع به یک مشکل حجم حافظه بر میگردد که نتیجه پر شدن و سراریز شدن اطلاعات از حافظه میگردد و در اصطلاح به آن tack Overflow میگویند،که میتواند عواقب متنوعی از جمله کرش کردن برنامه ، انجام محاسبات اشتباه ، و حتی نفوذ و شکسته شدن کد بینجامد. البته این مورد در پایتون نیازی به بررسی و مشکل زدایی ندارد و در زبان هایه سطح پایین تر مثل c این مشکل به وجود میاید و باید مورد بررسی قرار گیرد .

**4.3:**

در این کد ما شاهد یک فانکشن هستیم که این فاکشن مقادیری را محاسبه میکند.نکته قابل توجه اینجاست که این مقادیر باید در انتها توسط دستور return باز گردند تا قابل استفاده باشند. به همین منظور را با تغییر این دستور بازنویسی می کنیم.

**4.4:**

در خط چهارم شاهد دستور اصلی هستیم . این تابع مقدار هر یک از اعضایه لیست را به توان دو میرساند سپس این مقادیر را با هم جمع میکند و نتیجه نهایی را به عنوان خروجی باز میگرداند.

احمدرضا ذبیحی به نام خدا ش.د: ۴۰۰۵۲۱۳۴۲

**4.5:**

در ابتدا این تابع باید 3 مقدار ساعت ، دقیقه و مقدار را دریاف کند.در مرحله بعد تعداد ساعات را ضرب در تعداد ثانیه هر ساعت میکند و مساوی متغییر مربوطه قرار میدهد. در مرحله بعد همین عملیات را دقایق اجرا میکند. در انتها مجموع این ثانیه ها را برمیگرداند.

همچنین در ادامه مقدار 3 ورودی را دریافت کرده و سپس تابع را بر روی انها اعمال و نتیجه را نمایش میدهیم.

**4.6:**

در این مسعله جهت تعویض مشکل ما یک ارگومان بغیر از تعداد ارگومان هایه دلخواه به مجموعه ورودی تابع اضافه میکنیم.نتیجه این عمل این است که وقتی تابع را بدون ورودی فراخانی میکنیم اضافه کردن حداقل یک ورودی را اجباری میکند و دیگر ارور مخرج 0 را نمیگیریم .

**4.7:**

ابتدا کتاب خانه تایم را وارد کرده سپس زمان با فراخانی تابع مذکور در دستور پرینت چاپ میکنیم.

**4.8:**

بر اساس دستورات گفته شده در استفاده از دستور round در قسمت دوم ورودی عدد وارد شده مقدار اعشاری است که میخواهیم تقریب بزنیم به شکلی که در کد میبیینید.

**4.9:**

ابتدا تابعی با نام فارنهایت ایجاد کرده که یک مقدار ورودی دمایه سیلیسیوس را دریافت میکند .این مقدار ار با استفاده از فرمول تبدیل به فارنهایت میکند سپس مقدار را برمیگرداند.در نهایت با استفاده از حلقه فور این تابع را برایه دمایه 0 تا 100 درجه فارنهایت نمایش میدهیم.

احمدرضا ذبیحی به نام خدا ش.د: ۴۰۰۵۲۱۳۴۲

**4.10:**

کتابخانه رندوم را فراخانی میکنیم.جواب را مساوی عددی رندوم بین 1 تا 1000 قرار میدهیم .مقدار اولیه حدس را از کابر دریافت میکنیم. برابری حدس و عدد را بررسی میکنیم.اگر جواب مساوی ورودی بود پیام تبریک را چاپ میکنیم در غیر این صورت تا زمانی که جواب و ورودی برابر نباشد برنامه زیر را تکرار میکنیم.اگر جواب مساوی 0 بود برنامه را تمام میکنیم در غیر این صورت اگر ورودی از حدس بزگتر بود بزرگی را چاپ و مقدار جدید را دریافت کرده و همین مراحل را دوباره تکرار میکنیم در غیر این صورت عبارت کوچکتری را چاپ کرده و حدس جدید را دریافت کرده و مراحل قبل را تکرار میکنیم.

**4.11:**

در ابتدایه برنامه قبلی یک متغیر شمارش اضافه میکنیم که بعد از هر بار دریافت ورودی جدید یک عدد به مقدار ان اضافه میشود. در انتها وقتی کاربر به جواب درست رسید در جواب اگر مقدار شمارشگر کمتر از 10 بود پیغام شماره 1 و در غیر این صورت پیغام شماره 2 را چاپ میکنیم.

**4.12:**

در ابتدا دو کتابخانه رندوم و تایم را وارد میکنیم .(رندوم برایه دریافت حرکات تصادفی و تایم جهت نگه داشتن برنامه و اجرایه ان در هر ثانیه 1 بار) .

تابع اول را با نام لاک پشت که یک ورودی عددی با عنوان موقعیت میگیرد بدین شکل تعریف میکنیم. یک عدد رندوم بین 1 تا 10 انتخاب میکنیم و ان را مساوی متغیر نوع حرکت قرار میدهیم.اگر نوع حرکت به بین 1تا 5 یا 8 تا 10 باشد به تعداد تعریف شده به مقدار پوزیشن اضافه میکنیم و مقدار موقعیت را برمیگردانیم.اگر نوع حرکت برابر 6 یا 7 باشد از موقعیت 6 تا کم میکنیم و اگر موقعیت کمتر از خانه 1 شود ان را مساوی 1 قرار میدهیم و این را برمیگردانیم.

تابع دوم با عنوان خرگوش دقیقا با الگوریتم تابع لاک پشت پیش میرود با تفاوت کوچک در مقدار خانه ها و انواع حرکات.

موقعیت اولیه لاک پشت و خرگوش را در خانه 1 قرار میدهیم. سپس برنامه اصلی در یک حلقه شروع میشود که با شرط اگر لاک پشت یا خرگوش از خانه 70 رد شدند میشکند و خارج میشود .

احمدرضا ذبیحی به نام خدا ش.د: ۴۰۰۵۲۱۳۴۲

**4.12(ادامه):**

در ابتدا با فراخانی تابع تایم اجرایه حلقه را به هر ثانیه یکبار محدود میکنیم.سپس با فراخانی توابع لاکپشت و خرگوش موقعیت جدید انهارا در یافت میکنیم.در ادامه اگر موقعیت هر دو یکسان بود خط را با این فرمول چاپ میکنیم:( مقدار 70 را منهایه خانه ای که انها در ان قرار دارد میکنیم و به تعداد به دست امده خط چین میگزاریم سپس نوشته cuch!! را چاپ میکنیم و پس از ان به تعداد یکی کمتر از موقعیت ان ها خط چین چاپ میکنیم.)(هر یک خطچین به معنی یک خانه از 70 خانه است و خانه ای که خرگوش یا لاکپشت یا هردویه انها قرار دارند با عبارات و ایموجی ها نشان داده میشود.)در انتهایه خط موقعیت لاکپشت و خرگوش را به صورت عددی گزارش کرده و به خط بعد میرویم.اگر موقعیت این دو یکسان نبود و خرگوش جلوتر بود . خط جدید را به این ترتیب چاپ میکنیم:( مقدار70منهایه موقعیت خرگوش خط چین چاپ میکنیم.سپس خرگوش را نشان داده بعد به تعداد اختلاف خانه بین خرگوش و لاک پشت خطچین چاپ کرده سپس لاک پشت را نشان میدهیم و پس از ان به تعداد خانه هایه باقی مانده خط چین رسم میکنیم.)در انتهایه خط موقیت این دورا به صورت عددی گزارش میدهیم.اگر طی همین شرایط لاک پشت جایه خرگوش بود دقیقا همین حرکات را انجام میدهیم و فقط جای پوزیشن هایه خرگوش و لاک پشت را جابجا میکنیم.

در انتها وقتی حلقه بشکند اگر هر دو باهم در یک خانه باشند نوشته برابری ظاهر و اگر هر کدام زود تر به خط پایان برسد چاپ میکنیم که ان حیوان برنده شده است .

**4.13:**

تابع ضرب را با دریافت یک ورودی به شگل لیست به این ترتیب درست میکنیم:(متغییر y را برابر 1 قرار میدهیم زیرا ضرب هر عدد در 1 برابر خود عدد است سپس به ازایه هر یک اعضایه لیست ان در y ضرب میکنیم و در نهایت مقدار نهایی را برمیگردانیم.)سپس یک ورودی به شکل لیست از کاربر دریافت میکنیم. این لیست را با تابع بالا فراخانی میکنیم و نتیجه را نمایش میدهیم.

**4.14:**

تابع رندوم را وارد میکنیم.2 عدد به صورت رندوم در رنج 0 تا 9 انتخاب کرده و مقدار ضرب انهارا محاسبه میکنیم و سپس نتیجه را از دانش اموز میخاهیم و تا زمانی که دانش اموز جواب درست را ندهد به دریافت ورودی ادامه داده و در نهایت که به جواب درست رسید اورا تشویق کرده و این عملیات را دوباره تکرار میکنیم.

احمدرضا ذبیحی به نام خدا ش.د: ۴۰۰۵۲۱۳۴۲

**4.15:**

با اضافه کردن یک متغیر جدید به نام luck که یک مقدار رندوم بین 1 تا 3 دارد مشکل را حل میکنیم. هر بار که این متغیر یک مقدار متفاوت دارد با بررسی ان توسط شروط پیغام ها عوض میشود.

**4.16:**

در ابتدایه برنامه یک متغیر جدید قرار میدهیم که یکی از دو مقدار 1 یا 2 را از کاربر دریافت میکند. سپس دو متغیر جدید با نام شروع و پایان میسازیم. اگر متغیر اولیه برابر 1 بود مقدار شروع را برابر 0 و پایان برابر 9 خواهد بود و در غیر این صورت شورع و پایان برابر 10 و 99 خواهد بود. سپس کافیست در برنامه قبلی در قسمت رنج انتخاب عدد این دو مقدار شروع و پایان را به عنوان رنج انتخابی وارد کنیم.

**4.17:**

در ابتدا کتابخانه رندوم را فراخانی میکنیم. تابع برابری را با دریافت دو مقدار که مقدار اول مقداریست مغییر و جواب دانش اموز و متغییر دوم برابر مقدار اصلی جواب است ایجاد میکنیم. این تابع دقیقا دستورات برابری و نتیجه کد قبلی هستند که برابری را مقایسه و نتیجه را گزارش میکنند.

سپس 4 تابع با عناوین جمع ،تفریق، ضرب ، و تقسیم ایجاد کرده که هر کدام 2 عدد به عنوان ورودی دریافت میکنند سپس عملیات مربوطه را انجام میدهند و نتیجه را در یک متغیییر ذخیره کرده و سپس از کاربر یک ورودی جدید به عنوان جواب میگیرند،اگر کاربر مقدار -90 را وارد کند تابع یک استرینگ را برمیگردانند. در غیر این صورت تابع برابری را برایه این مقادیر اجرا میکنند.(نکته :درتابع تقسیم اگر عدد دوم وارد شده برابر 0 باشد تابع یک مقدار رندوم جدید ایجاد کرده و ان را برابر مقدار جدید مخرج کسر قرار میدهد تا مشکل ریاضیاتی مخرج صفر رخ ندهد.)

سپس قسمت اصلی برنامه شروع میشود که در یک حلقه قرار دارد.این مرحله بر اساس کد تمرین قبلیست و لول انتخاب رنج اعداد را مشخص میکند و ورودی -1 حلقه را مشکند.سپس یک حلقه دیگر شروع میشود که از کابر میخواهد شماره نوع سوال که میخواهد حل کند را وارد کند. سپس یک حلقه جدید تشکیل میشود که دو عدد رندوم در ان شکل میگیرد.اگر کاربر مقادیر 1 تا 4را وارد میکرد هر یک از توابع جمع ،تفریق،ضرب یا تقسیم به تنهایی فراخانی و اجرا میشوند و اگر خروجی استرینگ go out از انها دریافت شود حلقه شکسته میشود و کاربر میتواند نوع سوال را تغییر دهد در غیر این صورت برنامه به پرسیدن سوال جدید ادامه میدهد.

احمدرضا ذبیحی به نام خدا ش.د: ۴۰۰۵۲۱۳۴۲

**4.17(ادامه):**

اگر کاربر نوع سوال 5 را انتخاب کند. یک لیست از توابع ایجاد میکنیم سپس یک تابع را به صورت رندوم از بین انها انتخاب کرده و انرا اجرا میکنیم مانند توابع قبلی با همان قوانین.دقت داشته باشید که این عمل انتخاب تابع به صورت رندوم در هر بار گردش حلقه رخ میدهد در نتیجه هر سوال نسبت به سوال قبلی رندوم است و میتواند متفاوت باشد.