



طراحی زبان‌های برنامه‌سازی

پاییز ۱۴۰۰

استاد: محمد ایزدی

گردآورندگان: علی عباسی، امیرعلی ابراهیم‌زاده

بررسی و بازبینی:

دانشگاه صنعتی شریف

دانشکده مهندسی کامپیوتر

مهلت ارسال: ۲ آذر

فصل دوم EOPL

تمرین سوم

به موارد زیر توجه کنید:

- (۱) برنامه‌های خود را به زبان Racket بنویسید.
- (۲) مهلت ارسال تمرین ساعت ۵۹ : ۲۳ روز ۲ آذر ۱۴۰۰ است.
- (۳) در مسائلی که نیاز به خروجی دادن است، پاسخ را چاپ نکنید بلکه آن را به عنوان حاصل تابع برگردانید.
- (۴) حتما تابع اصلی خود را در main بگذارید تا هنگام داوری به مشکل نخورید. منظور از تابع اصلی تابعی است که اجرای آن روی ورودی، خروجی مطلوب را نشان می‌دهد.
- (۵) جواب هر سوال برنامه نویسی را در یک فایل Racket با نام و فرمت زیر قرار دهید.

`[StudentID]_[ProblemNumber].rkt`

- (۶) در نهایت تمام فایل‌های Racket را در یک فایل زیپ قرار داده و با نام `HW3_StudentID` در سامانه کوثرآپلود کنید.
- (۷) هرگونه سوالی راجع به تمرین را در زیر پست مربوطه در کوثرای درس مطرح کنید.
- (۸) در مجموع تمامی تمارین ۷ روز مهلت تاخیر مجاز دارید و پس از تمام شدن این تاخیرهای مجاز به ازای هر روز ۱۰ درصد از کل نمره تمرین شما کم می‌شود.
- (۹) لطفا تمرین‌ها را از یکدیگر کپی نکنید. در صورت وقوع چنین مواردی مطابق با سیاست درس رفتار می‌شود.

سوالات (۱۰۰ نمره)

۱. (۱۵ نمره) Abstract Syntax و Concrete Syntax را تعریف کرده و با ارائه یک مثال تفاوت آن‌ها را بیان کنید.
۲. (۲۰ نمره) نوع داده صف را در نظر بگیرید و برای هر یک از توابع زیر یک توصیف نوشته و سپس این توابع را پیاده‌سازی کنید.

- empty-queue – Return a new empty queue.
- enqueue – Add an item to the queue.
- dequeue – Remove an item from the queue.
- peek – Gets the element at the front of the queue without removing it.
- isfull – Checks if the queue is full.
- isempty – Checks if the queue is empty.

نحوه‌ی پیاده‌سازی و جزئیات آن (همانند ورودی‌های مورد نیاز توابع یا چگونه در نظر گرفتن طول برای این صف) بر عهده خودتان است.

۳. (۲۰ نمره) نمایش درختی *Diff-tree* را برای اعداد صحیح در نظر بگیرید، که *Diff-tree* یک لیست با گرامر زیر است:

$Diff-tree ::= (one) \mid (diff \ Diff-tree \ Diff-tree)$

که عددی که هر درخت نشان می‌دهد از این طریق به دست می‌آید:

$value\ of\ (one) = 1$

$value\ of\ (diff\ t_1\ t_2) = value\ of\ t_1 - value\ of\ t_2$

الف) نشان دهید در این سیستم نمایش اعداد، برای هر عدد بی‌نهایت نمایش مختلف می‌توان متصور شد.
ب) چهار تابع $zero?$ و $is - zero?$ و $successor$ و $predecessor$ را برای این نمایش پیاده‌سازی کنید.
ج) تابع $diff-tree-plus$ را به گونه‌ای پیاده‌سازی کنید که دو $Diff-tree$ دریافت کند و $Diff-tree$ مربوط به حاصل جمع عدد این دو درخت را در زمان ثابت ($O(1)$) خروجی دهد.

۴. (۲۰ نمره) نوع داده‌ی محیط ($environment$) را در نظر بگیرید. به روش procedural representation این نوع داده را به گونه‌ای پیاده‌سازی کنید که ویژگی‌های زیر را نیز داشته باشد:

- هیچ‌گاه ارور ندهد و در صورت نبودن داده در آن، مقدار $'not-set-variable'$ برگردانده شود.
- تابع $has-binding?$ را به گونه‌ای پیاده‌سازی کنید که با گرفتن یک محیط و یک متغیر، در صورت موجود بودن آن متغیر در محیط $\#t$ و در غیر این صورت $\#f$ برگرداند.
- تابع $union$ را پیاده‌سازی کنید. که با گرفتن دو محیط، اجتماع حاصل از دو محیط را برگرداند. و در صورت وجود یک متغیر در هر دو محیط، مقدار آن را از محیط دوم خروجی بدهد.

۵. (۲۵ نمره) در این سوال می‌خواهیم ذخیره‌ی اعداد صحیح نامنفی به روش Bignum را پیاده‌سازی کنیم. در این روش ابتدا یک پایه در نظر گرفته می‌شود (N) و سپس هر عدد را به شکل لیستی از اعداد بین ۰ و $N - 1$ ذخیره می‌کنیم که عضو i ام این لیست نمایانگر ضریب N^i تجزیه‌ی عدد به پایه‌ی N است (می‌توانید این گونه تصور کنید که عدد را به مبنای N می‌نویسیم و ضرایب را ذخیره می‌کنیم). توابع $successor$ و $predecessor$ را برای این نوع نمایش پیاده‌سازی کنید. تابع $zero$ ، عدد N را به عنوان ورودی می‌گیرد.