

دانشگاه علم و صنعت

دانشکده مهندسی کامپیوتر

درجه تحصیلی: کارشناسی

گزارشکار تکلیف2 OS

گردآورنده:

پرنیان شاکریان - 99400064

استاد:

دکتر انتظاری

سال تحصیلی: اسفند 1401

تمام کتابخانه ها و headerهای مورد نیاز ما برای تمرین 2 به طور کلی و برای آسانی کار اینجا نوشته و اشاره شده است.

#include <assert.h>

#include <stdio.h>

#include <pthread.h>

#include <stdlib.h>

#include <unistd.h>

#include <sys/types.h>

#include <sys/stat.h>

#include <sys/wait.h>

#include <string.h>

برای ساخت makefile ها از لینک های زیر کمک گرفته ام.

<https://youtu.be/bPMDEdjKBWA>

<https://youtu.be/_r7i5X0rXJk>

سوال 3: یه برنامه بنویسید که روت یک درخت را داریم و به دنبال عدد x میگردیم. با استفاده از multi-process مسیر root تا node که حاوی عدد x هست را چاپ کنید مسئله را به صورت divide and conquer حل کنید.

به طور کلی این برنامه یک الگوریتم جستجو را برای ساختار داده درختی پیاده سازی می کند. درخت به عنوان یک گره ساختاری نشان داده می شود که حاوی یک مقدار و آرایه ای از اشاره گرها به گره های فرزند خود است. این برنامه یک درخت با یک گره ریشه و سه سطح از گره های فرزند ایجاد می کند و سپس با استفاده از تابع search\_tree به دنبال یک گره با یک مقدار خاص می شود. تابع جستجو دارای چهار آرگومان است: یک اشاره گر به گره فعلی مورد جستجو، مقدار مورد جستجو، یک آرایه برای ذخیره مسیر از ریشه به گره با مقدار، و سطح فعلی درخت مورد جستجو.

تابع search\_tree به صورت بازگشتی درخت را با بررسی هر گره فرزند از گره فعلی برای مقدار جستجو می کند. اگر مقدار در یکی از گره های فرزند یافت شود، تابع مسیر آن گره را در آرایه مسیر ذخیره می کند و true را برمی گرداند. اگر مقدار در هیچ یک از گره های فرزند یافت نشد، تابع false را برمی گرداند.

در تابع main، برنامه یک درخت ایجاد می کند، مقداری را برای جستجو تعیین می کند و تابع search\_tree را برای جستجوی آن مقدار اجرا می کند. اگر مقدار پیدا شد، برنامه مسیر را از ریشه به گره با مقدار چاپ می کند. اگر مقدار پیدا نشد، برنامه پیامی را چاپ می کند که نشان می دهد مقدار پیدا نشد.

اگر بخواهیم خط به خط کد را توضیح دهیم تابع اول یک مقدار صحیح را به عنوان ورودی می گیرد و یک اشاره گر را به ساختار Node جدید برمی گرداند. این تابع یک نمونه Node جدید با استفاده از malloc ایجاد کرده تا حافظه را به صورت پویا به آن اختصاص دهد. این تابع، فیلد مقدار گره تازه ایجاد شده را روی مقدار ورودی تنظیم می کند و تمام اشاره گرهای فرزند خود را به NULL مقداردهی اولیه می کند.

سپس از خط 21 تابع از طریق درخت یک مقدار صحیح داده شده را جستجو می کند. چهار آرگومان نیاز دارد که به ان اشاره شد این تابع با بررسی NULL بودن گره فعلی شروع می شود. در این صورت، تابع بلافاصله false را برمی گرداند سپس، تابع مقدار گره فعلی را در آرایه مسیر در سطح فعلی ذخیره می کند. سپس بررسی می کند که آیا مقدار گره فعلی با مقدار ورودی مطابقت دارد یا خیر. اگر چنین است، تابع true را برمی گرداند. اگر مقدار ورودی در گره فعلی یافت نشد، تابع یک متغیر بولین یافت شده را روی false تنظیم می کند و روی هر فرزند گره فعلی حلقه می زند. اگر یک گره فرزند داده شده NULL نباشد، تابع به صورت بازگشتی جستجو را از طریق فرزند فراخوانی می کند. اگر جستجو از طریق فرزند true را برگرداند، این نشان می دهد که مقدار ورودی در جایی در زیر درخت آن پیدا شده است، بنابراین پرچم یافت شده روی true تنظیم می شود و حلقه روی فرزندان دیگر با استفاده از break خاتمه می یابد. if نهایی بررسی می‌کند که آیا مقدار پیدا شده است، همانطور که با بولی پیدا شده نشان داده شده است. اگر پیدا نشد، ورودی مسیر [سطح] روی -1 تنظیم می شود. سپس تابع پیدا شده را برمی گرداند. این تابع یک درخت ساده تنظیم می کند و سپس تابع search\_treeرا با این درخت به عنوان ورودی فراخوانی می کند تا مشخص کند که آیا مقدار 2 در درخت یافت می شود یا خیر. برنامه با فراخوانی create\_node برای تخصیص حافظه برای گره های جدید و تنظیم مقادیر آنها شروع می شود. پس از این، برنامه یک مسیر آرایه عدد صحیح را با NUM\_CHILDREN (که در حال حاضر به عنوان 3 تعریف می‌شود) مقداردهی می‌کند تا مسیری را که از طریق درخت طی شده ذخیره کند. برنامه مقدار 2 را به مقدار متغیر صحیح اختصاص می‌دهد و تابع search\_tree را برای ذخیره مسیر پیمایش فراخوانی می‌کند و با تمام 0ها مقدار دهی اولیه می‌شود.( عدد صحیح 0 به عنوان شاخص مسیر شروع)

دستور if در کد بررسی می کند که آیا تابع search\_tree() true برمی گرداند یا خیر. اگر مقدار را بیابد، برنامه مسیری را که در درخت از ریشه تا گره هدف طی شده است چاپ می کند. اگر مقدار پیدا نشد، برنامه پیامی مبنی بر یافت نشدن آن خروجی می دهد. پس از اتمام، برنامه 0 را برمی گرداند.

خروجی مثال من:

