



دانشگاه جهرم

## استفاده از پردازش‌های موازی در سیستم عامل لینوکس

نگارش:

امیررضا ارجمند

استاد:

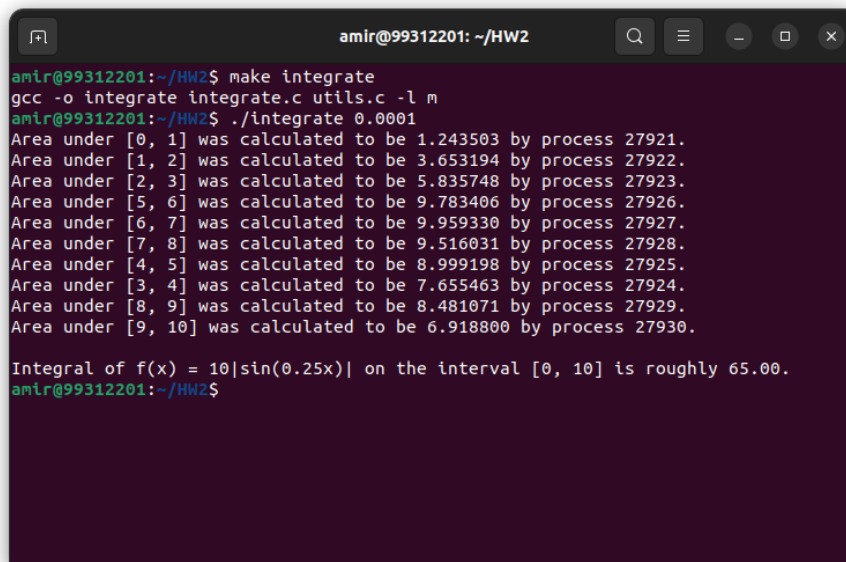
دکتر محمدجواد پارسه

شماره دانشجویی:

۹۹۳۱۲۲۰۱

## ۱. توضیحاتی درمورد برنامه

این برنامه مقدار تقریبی  $\int_0^{10} 10|\sin 0.25x| dx$  را با استفاده از ۱۰ پردازنده موازی و به کمک روش دوزنقه‌ای محاسبه می‌کند. کدهای مورد نیاز برای ساختن و اجرای این پروژه در پوشه SRC ذخیره شده است. برای ساختن پروژه می‌توانیم از دستور `make integrate` استفاده کنیم (در صورت نیاز باید ابتدا ابزار `make` را نصب کنیم). سپس با استفاده از دستور `./integrate` و مشخص کردن  $dx$  به عنوان آرگمان دوم خط دستور، انتگرال را محاسبه کنیم.



```
amir@99312201: ~/HW2
amir@99312201:~/HW2$ make integrate
gcc -o integrate integrate.c utils.c -l m
amir@99312201:~/HW2$ ./integrate 0.0001
Area under [0, 1] was calculated to be 1.243503 by process 27921.
Area under [1, 2] was calculated to be 3.653194 by process 27922.
Area under [2, 3] was calculated to be 5.835748 by process 27923.
Area under [3, 4] was calculated to be 7.655463 by process 27924.
Area under [4, 5] was calculated to be 8.999198 by process 27925.
Area under [5, 6] was calculated to be 9.783406 by process 27926.
Area under [6, 7] was calculated to be 9.959330 by process 27927.
Area under [7, 8] was calculated to be 9.516031 by process 27928.
Area under [8, 9] was calculated to be 8.481071 by process 27929.
Area under [9, 10] was calculated to be 6.918800 by process 27930.

Integral of f(x) = 10|sin(0.25x)| on the interval [0, 10] is roughly 65.00.
amir@99312201:~/HW2$
```

توجه داشته باشید که  $dx$  باید چنان باشد که ضربی از آن دقیقاً برابر با ۱ شود و نباید از ۱ هم بیشتر باشد. هرچه مقدار  $dx$  کوچکتر باشد مساحت محاسبه شده توسط هریک از پردازنده‌ها دقیق‌تر خواهد بود اما از آنجایی که تمام پردازنده‌ها عددی صحیح را به پردازنده‌ی والد برمی‌گردانند، بخش اعشاری عدد محاسبه شده قطع شده و نتیجه نهایی دقیق نخواهد بود.

هدف از این گزارش کار، یادگیری کار با پردازنده‌های فرزند و والد و ارسال اطلاعات میان آنهاست. در صورت وجود بیش از یک پردازنده در کامپیوتر، پردازنده‌ها می‌توانند به صورت همزمان محاسبات خود را انجام دهند و سریع‌تر به نتیجه نهایی برسند. این مساله را می‌توان با نوشتن همین برنامه به صورت تک پردازنده‌ای و مقایسه سرعت آنها مشاهده کرد.

```
amir@99312201: ~/HW2
amir@99312201:~/HW2$ make
gcc -o integrate integrate.c utils.c -l m
gcc -o integrate_sequential integrate_sequential.c utils.c -l m
amir@99312201:~/HW2$ time ./integrate 0.000001 > /dev/null

real    0m0.146s
user    0m0.518s
sys     0m0.016s
amir@99312201:~/HW2$ time ./integrate_sequential 0.000001 > /dev/null

real    0m0.413s
user    0m0.405s
sys     0m0.005s
amir@99312201:~/HW2$
```

## ۲. توضیحاتی درمورد توابع و header file های استفاده شده

- ۲/۱. `<stdlib.h>`: یک کتابخانه عمومی در زبان C است که حاوی توابعی برای تخصیص حافظه، کنترل اجرای فرایندها، تبدیل نوع متغیرها و دیگر کارهای رایج در برنامه می باشد. این کتابخانه مانند `<stdio.h>` تقریباً در تمامی فایل های C استفاده شده.
- ۲/۲. `<unistd.h>`: این کتابخانه غالباً شامل توابعی برای استفاده از System Call هایی مانند `write`، `read`، `pipe`، `fork` و غیره می باشد.
- ۲/۳. `<sys/wait.h>`: در این فایل توابع و Macro هایی برای انتظار برای پایان فرایند فرزند، گرفتن حالت خروجی آن و استفاده از این اطلاعات تعریف شده.
- ۲/۴. `WIFEXITED`: این دستور در صورت خارج شدن فرایند به صورت طبیعی مقدار ۱ و در غیر این صورت مقدار ۰ را بر می گرداند. در این برنامه تمام فرایندهای فرزند باید به درستی خاتمه یابند و برای تشخیص این مساله از این دستور استفاده می شود.
- ۲/۵. `WEXITSTATUS`: این دستور تمامی مقادیر بازگشتی از فرایند فرزند که شامل `status code` و `flag` ها است را دریافت کرده و فقط `status code` را بر می گرداند.
- ۲/۶. `fork`: این فراخوان سیستمی برای تولید یک فرایند فرزند استفاده می شود به طوری که فرایند والد و فرزند از خط بعد به صورت همزمان اجرا می شوند. مقدار بازگشتی این تابع برای فرایند فرزند ۰ و برای فرایند والد `id` فرایند فرزند می باشد.
- ۲/۷. `exit`: این فراخوان سیستمی برای خروج یک فرایند استفاده می شود. ورودی این تابع `status code` فرایند می باشد که به فرایند والد ارسال می شود.
- ۲/۸. `wait`: منتظر اتمام یکی از فرایندهای فرزند می ماند. ورودی این تابع آدرس یک متغیر `integer` است که مقدار بازگشتی توسط فرایند فرزند در آن ذخیره می شود.
- همچنین تعدادی تابع کمکی در این برنامه استفاده شده که در `utils.c` ذخیره شده اند. توضیحات استفاده از هریک از توابع در همان تابع به صورت کامنت نوشته شده.

### ۳. نحوه ایجاد پردازنده‌ها

این برنامه شامل یک پردازنده والد و ۱۰ پردازنده فرزند می‌باشد. با فراخوانی نام برنامه در محیط terminal می‌توانیم فرایند والد را اجرا کنیم. سپس فرایندهای فرزند توسط فرایند والد و با تابع سیستمی fork ساخته می‌شوند.

### ۴. نحوه کامپایل برنامه

برای این کار از ابزار make استفاده شده که در اکثر پروژه‌های C به عنوان ابزاری برای ساخت و مدیریت کامپایل پروژه استفاده می‌شود. این ابزار به صورت پیش‌فرض در سیستم عامل Ubuntu نصب نیست ولی می‌توان با دستور `sudo apt install make` به سادگی آن را نصب کرد. سپس با دستور `make integrate` می‌توانیم کد C را کامپایل کنیم. همچنین می‌توانیم با دستور `make integrate_sequential` کدی مشابه با `integrate.c` که فقط از یک پردازنده استفاده می‌کند را کامپایل کنیم و سرعت این دو را با هم مقایسه کنیم.

### ۵. اجرای برنامه و نتیجه گیری

انتگرال این تابع در بازه  $[0, 10]$  تقریباً برابر با  $74/045$  می‌باشد. از آنجایی که هریک از پردازنده‌ها مقداری صحیح به والد خود بر می‌گردانند و از بخش اعشاری صرف نظر می‌کنند فرایند والد عدد ۶۵ را بدست می‌آورد. می‌توانیم با جمع زدن مقادیر محاسبه شده توسط هر پردازنده به صورت دستی مقدار دقیق‌تری را محاسبه کنیم.

```
amir@99312201: ~  
amir@99312201:~$ python3  
Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] on linux  
Type "help", "copyright", "credits" or "license()" for more information.  
>>> 1.243503 + 3.653194 + 5.835748 + 9.783406 + 8.999198 + 9.516031 + 8.481071 +  
9.959330 + 7.655463 + 6.918800  
72.045744  
>>>
```

```
amir@99312201: ~/HW2  
amir@99312201:~/HW2$ make integrate  
gcc -o integrate integrate.c utils.c -l m  
amir@99312201:~/HW2$ ./integrate 0.0001  
Area under [0, 1] was calculated to be 1.243503 by process 2799.  
Area under [1, 2] was calculated to be 3.653194 by process 2800.  
Area under [2, 3] was calculated to be 5.835748 by process 2801.  
Area under [3, 4] was calculated to be 9.783406 by process 2804.  
Area under [4, 5] was calculated to be 8.999198 by process 2803.  
Area under [7, 8] was calculated to be 9.516031 by process 2806.  
Area under [8, 9] was calculated to be 8.481071 by process 2807.  
Area under [6, 7] was calculated to be 9.959330 by process 2805.  
Area under [3, 4] was calculated to be 7.655463 by process 2802.  
Area under [9, 10] was calculated to be 6.918800 by process 2808.  
  
Integral of  $f(x) = 10|\sin(0.25x)|$  on the interval  $[0, 10]$  is roughly 65.00.  
amir@99312201:~/HW2$
```