



## دستورکار جلسه پنجم

۱. برنامه ای بنویسید که در آن:

- والد به تعداد MAX\_CHILD فرزند داشته باشد.
  - بین والد و هر فرزند آن یک خط لوله برقرار باشد.
  - در ابتدا، متغیر total=0 در والد مقداردهی می شود.
  - هر فرزند در یک حلقه اجرا می شود و با فاصله ی زمانی t ثانیه عددی تصادفی بین ۰ تا ۱۰ ایجاد کرده، برای والد می فرستد.
  - t نیز مقداری تصادفی است، در اینجا می توان از t=1 استفاده کرد.
  - والد مقدار ارسال شده توسط هر فرزند را دریافت کرده و با total جمع می کند.
  - والد فرزندان خود را بررسی می کند و در هر بار بررسی مقدار کنونی total را نمایش می دهد.
  - هر گاه مقدار total از MAXIMUM (برای مثال MAXIMUM=100) بیشتر شود، والد همه فرزندان را از بین می برد و خود نیز به پایان می رسد
- راهنمایی سؤال ۱: برای از بین بردن فرزندان از kill(pid, SIGKILL) استفاده کنید

۲. برنامه ای بنویسید که در آن:

- والد به تعداد MAX\_CHILD فرزند دارد و به هر فرزند یک id اختصاص داده می شود (مقداری بین ۰ و MAX\_CHILD-1)
  - هر یک از فرآیندهای فرزند به اندازه t ثانیه اجرا می شوند و پس از آن بایستی منتظر بمانند تا سایر فرآیندها نیز به همان نقطه برسند، در اینجا t مقداریست بین ۱ تا ۱۰.
  - برای ایجاد هماهنگی بین فرآیندها فرآیند والد مقدار MAX\_CHILD-1 یعنی شناسه آخرین فرزند را در یک خط لوله نامگذاری شده می نویسد.
  - هر یک از فرآیندهای فرزند پس از رسیدن به نقطه ی هماهنگی، مرتباً مقدار موجود در خط لوله را بررسی می کند و بسته به مقدار آن یکی از شرایط زیر ایجاد می شود:
- (a) در صورتی که مقدار برابر ۰ باشد، برنامه می تواند از نقطه ی هماهنگی عبور کند.
- (b) در صورتی که مقدار بزرگتر از ۰ و برابر با id فرآیند است، از مقدار موجود در خط لوله یک واحد کاسته شده و فرآیند همچنان در نقطه ی هماهنگی منتظر می ماند، یعنی تا دریافت صفر در pipe، خواندن از pipe را ادامه می دهد.

راهنمایی سؤال ۲:

- جهت اطلاع هر client از id خود، یک متغیر (مثلاً child\_id) در والد قبل از ایجاد فرزندان تعریف کنید. سپس در حلقه for ایجاد فرزندان این متغیر را برای هر فرزند ایجادشده با مقدار شمارنده حلقه مقداردهی کنید. بدین صورت این متغیر در هر فرزند مقدار شناسه خود را می گیرد.
- برای اجرای قسمت a کافیست هر فرزند اگر مقداری که از pipe خوانده برابر صفر بود، صفر را دوباره در pipe بنویسد (برای اطلاع بقیه پروسس ها از رسیدن به نقطه هماهنگی) و خودش خاتمه یابد.
- برای اجرای قسمت b کافیست هر فرزند اگر مقداری که از pipe خوانده برابر شناسه خود (child\_id) بود، یک واحد از این مقدار کم کرده و آن را دوباره در pipe بنویسد.
- اگر مقداری که از pipe خوانده شده برابر صفر یا شناسه خود پروسس نبود، پروسس دوباره به خواندن pipe ادامه می دهد و مراحل a یا b را انجام می دهد.
- به منظور اینکه بتوانیم نحوه اجرای برنامه را دنبال کنیم log های زیر را در کد اضافه کنید:
- در شروع اجرای هر فرزند عبارت child id starts چاپ شود (id شناسه آن فرزند است که والد به او اختصاص داده است).
- بعد از اینکه هر فرزند مقدار pipe را خواند خط عبارت child id pipevalue: n چاپ شود (id شناسه فرزند و n مقداری است که از pipe خوانده است).
- هر پروسس وقتی به نقطه هماهنگی رسید قبل از خاتمه عبارت child id reach the point را چاپ کند (id شناسه فرزند است).
- والد پس از آنکه همه فرزندان خاتمه یافتند عبارت The program finished successfully را چاپ کند.