

## استک و هیپ در برنامه نویسی

در برنامه نویسی، مدیریت حافظه برای ذخیره سازی و دسترسی موثر به داده ها بسیار مهم است. دو نوع متداول تخصیص حافظه، هیپ و استک هستند. درک تفاوت بین آنها برای نوشتن کد خوب ضروری می باشد.

### حافظه هیپ چیست؟

حافظه هیپ قسمتی از حافظه است که در حین اجرای برنامه به صورت پویا تخصیص داده می شود. برای ذخیره سازی داده هایی که فراتر از محدوده یک تابع یا بلوک باقی می ماند استفاده می شود. حافظه هیپ توسط برنامه نویس مدیریت می شود و معمولاً از طریق اشاره گرها قابل دسترسی است. برای مثال در زبان برنامه نویسی سی پلاس پلاس با استفاده از کلمه و سینتکس نیو و فری به این حافظه دسترسی مستقیم داشت. این امکان تخصیص حافظه پویا را فراهم می کند، به این معنی که حافظه را می توان در زمان اجرا در صورت نیاز تخصیص داد و آزادسازی کرد.

### حافظه استک چیست؟

حافظه استک قسمتی از حافظه است که برای ذخیره متغیرهای محلی و اطلاعات فراخوانی تابع استفاده می شود. هر بار که یک تابع فراخوانی می شود، یک جدول استک جدید ایجاد می شود که شامل پارامترهای تابع، آدرس برگشتی و متغیرهای محلی است. حافظه استک به طور خودکار توسط کامپایلر مدیریت می شود و از ساختار Last In, First Out (LIFO) پیروی می کند.

### موارد استفاده از حافظه هیپ و استک:

حافظه هیپ معمولاً برای تخصیص حافظه برای ساختارهای داده با اندازه پویا مانند آرایه ها، لیست های پیوندی و اشیاء استفاده می شود. از آنجایی که اندازه این ساختارهای داده ممکن است در زمان کامپایل مشخص نباشد، حافظه هیپ امکان انعطاف پذیری در تخصیص حافظه را فراهم می کند. حافظه استک برای ذخیره متغیرهای محلی و اطلاعات فراخوانی تابع استفاده می شود. استک سریع و کارآمد است زیرا تخصیص حافظه به طور خودکار توسط کامپایلر مدیریت می شود. با این حال، اندازه حافظه استک محدود است و استفاده بیش از حد می تواند منجر به خطاهایی در استک شود.

### تفاوت بین حافظه استک و هیپ:

#### ۱. تخصیص:

- حافظه هیپ: به صورت پویا توسط برنامه نویس با استفاده از توابعی مانند malloc() و free() تخصیص داده می شود.

- حافظه استک: به طور خودکار توسط کامپایلر مدیریت می شود.

#### ۲. مدیریت حافظه:

- حافظه هیپ نیاز به مدیریت صریح توسط برنامه نویس از جمله تخصیص و آزادسازی حافظه دارد.

- حافظه استک به طور خودکار توسط کامپایلر مدیریت می شود و خطر نشت حافظه و سایر خطاهای مربوط به حافظه را کاهش می دهد.

### ۳. دسترسی:

- حافظه هیپ از طریق اشاره گرها قابل دسترسی است و امکان تخصیص و توزیع پویا حافظه را فراهم می کند.

- حافظه استک مستقیماً با استفاده از نام متغیرها در محدوده یک تابع قابل دسترسی است.

### متغیرها در سی پلاس پلاس و تخصیص حافظه آنها:

در سی پلاس پلاس، متغیرها بسته به نحوه اعلان آنها، می توانند حافظه را بر روی هیپ یا استک قرار دهند:

- متغیرهایی که با استفاده از کلمه کلیدی نیو اعلان شده اند، حافظه روی هیپ تخصیص داده می شوند و باید به طور صریح با استفاده از کلمه کلیدی دیلیت آزادسازی داده شوند.

- متغیرهای محلی اعلان شده در یک تابع روی استک قرار داده می شوند و با بازگشت تابع به طور خودکار آزادسازی می شوند.

### نتیجه:

حافظه هیپ و استک نقش اساسی در مدیریت حافظه در برنامه نویسی دارند. درک تفاوت های آنها و استفاده مناسب برای نوشتن کد کارآمد و بدون خطا بسیار مهم است. با مدیریت دقیق تخصیص و آزادسازی حافظه، برنامه نویسان می توانند عملکرد کد خود را بهینه کنند و از مشکلات رایج مرتبط با حافظه جلوگیری کنند.

امیررضا محمدی یگانه