(2)

```
void DFS_NonRecursive(ALGraph *G, int v) {
if (v < 0 || v >= G->vexnum) {
printf("顶点索引错误！\n");
return;
}

bool visited[MAX_VERTEX_NUM] = {false}; // 访问标记数组
Stack stack;
initStack(&stack);

printf("从顶点 %c 开始的深度优先遍历（非递归）: ", G->vertices[v].data);

// 起始顶点入栈并标记为已访问
push(&stack, v);
visited[v] = true;
printf("%c ", G->vertices[v].data);

while (!isStackEmpty(&stack)) {
int current;
getTop(&stack, &current); // 查看栈顶元素，不出栈

// 查找当前顶点的第一个未访问邻接点
ArcNode *p = G->vertices[current].firstarc;
int nextVertex = -1;

while (p != NULL) {
if (!visited[p->adjvex]) {
nextVertex = p->adjvex;
break;
}
p = p->nextarc;
}

if (nextVertex != -1) {
// 找到未访问的邻接点，访问并入栈
visited[nextVertex] = true;
push(&stack, nextVertex);
printf("%c ", G->vertices[nextVertex].data);
} else {
// 当前顶点的所有邻接点都已访问，出栈
pop(&stack, &current);
}
}
```

```
printf("\n");
}
```

(3)