Zohaib ahmer

Fa20-bcs-038

{

"cells": [

{

"cell\_type": "code",

"execution\_count": 1,

"id": "da0baf45",

"metadata": {},

"outputs": [],

"source": [

"# LAB ACTIVITY 5 (SP19-BCS-042)\n",

"\n",

"def iterative\_deeping\_dfs(start,target):\n",

" depth=1\n",

" bottom\_reached=False\n",

" while not bottom\_reached:\n",

" result,bottom\_reached=iterative\_deepening\_dfs\_rec(start,target,0,depth)\n",

" if result is not None:\n",

" return result\n",

" \n",

" depth \*=2\n",

" print(\"increasing depth to \" + str(depth))\n",

" \n",

" \n",

" \n",

" "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "6d535fba",

"metadata": {},

"outputs": [],

"source": [

"# LAB ACTIVITY 5 (SP19-BCS-042)\n",

"\n",

"def iterative\_deepening\_dfs\_rec(node,target,current\_depth,max\_depth):\n",

" print (\"visiting node \"+ str(node[\"value\"]))\n",

" \n",

" if node[\"value\"] == target:\n",

" print(\"found the node we are looking for \")\n",

" \n",

" return node,True\n",

" \n",

" if current\_depth == max\_depth:\n",

" print(\"current maximum depth reached , returning...)\n",

" \n",

" if len(node[\"children\"]) > 0:\n",

" return None, False\n",

" \n",

" else:\n",

" return None,True\n",

" \n",

" bottom\_reached=Ture\n",

" for i in range(len(node[\"children\"])):\n",

" \n",

" result, bootom\_reached\_rec = iterative\_deepening\_dfs\_rec(node[\"children\"][i],target,current\_depth + 1 ,max\_depth)\n",

" \n",

" if result is not None:\n",

" return result,True\n",

" \n",

" bottom\_reached = bottom\_reached and bottom\_reached\_rec \n",

" \n",

" return None,bottom\_reached\n",

" \n"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "a3be8ecc",

"metadata": {},

"outputs": [],

"source": [

"# LAB ACTIVITY 5 (SP19-BCS-042)\n",

"\n",

"start={\n",

" \"value\":0,\"children\":[\n",

" {\" value \": 2, \"children\":[\n",

" {\" value \": 3, \"children\":[]},\n",

" {\" value \": 4, \"children\":[]}\n",

" ]},{\n",

" \" value \": 2, \"children\":[\n",

" {\" value \": 5, \"children\":[]},\n",

" {\" value \": 6, \"children\":[]}\n",

" \n",

" ]\n",

" \n",

" }\n",

" \n",

" ]\n",

"}\n",

"\n",

"print(iterative(start,6)[\"value\"]"

]

},

{

"cell\_type": "code",

"execution\_count": 1,

"id": "afe224de",

"metadata": {},

"outputs": [

{

"ename": "NameError",

"evalue": "name 'iterative' is not defined",

"output\_type": "error",

"traceback": [

"\u001b[1;31m---------------------------------------------------------------------------\u001b[0m",

"\u001b[1;31mNameError\u001b[0m Traceback (most recent call last)",

"\u001b[1;32m~\\AppData\\Local\\Temp\\ipykernel\_9436\\4012284006.py\u001b[0m in \u001b[0;36m<module>\u001b[1;34m\u001b[0m\n\u001b[0;32m 61\u001b[0m }\n\u001b[0;32m 62\u001b[0m \u001b[1;33m\u001b[0m\u001b[0m\n\u001b[1;32m---> 63\u001b[1;33m \u001b[0mprint\u001b[0m\u001b[1;33m(\u001b[0m\u001b[0miterative\u001b[0m\u001b[1;33m(\u001b[0m\u001b[0mstart\u001b[0m\u001b[1;33m,\u001b[0m\u001b[1;36m6\u001b[0m\u001b[1;33m)\u001b[0m\u001b[1;33m[\u001b[0m\u001b[1;34m\"value\"\u001b[0m\u001b[1;33m]\u001b[0m\u001b[1;33m)\u001b[0m\u001b[1;33m\u001b[0m\u001b[0m\n\u001b[0m",

"\u001b[1;31mNameError\u001b[0m: name 'iterative' is not defined"

]

}

],

"source": [

"# LAB ACTIVITY 5 (SP19-BCS-042)\n",

"\n",

"def iterative\_deeping\_dfs(start,target):\n",

" depth=1\n",

" bottom\_reached=False\n",

" while not bottom\_reached: \n",

" result,bottom\_reached=iterative\_deepening\_dfs\_rec(start,target,0,depth)\n",

" \n",

" if result is not None:\n",

" return result\n",

" \n",

" depth \*=2\n",

" print(\"increasing depth to \" + str(depth))\n",

" return None\n",

"\n",

"def iterative\_deepening\_dfs\_rec(node,target,current\_depth,max\_depth):\n",

" \n",

" print (\"visiting node \"+ str(node[\"value\"]))\n",

" \n",

" if node[\"value\"] == target:\n",

" print(\"found the node we are looking for \")\n",

" \n",

" return node,True\n",

" \n",

" if current\_depth == max\_depth:\n",

" print(\"current maximum depth reached , returning...\")\n",

" \n",

" if len(node[\"children\"]) > 0:\n",

" return None, False\n",

" \n",

" else:\n",

" return None,True\n",

" \n",

" bottom\_reached=Ture\n",

" for i in range(len(node[\"children\"])):\n",

" result, bottom\_reached\_rec = iterative\_deepening\_dfs\_rec(node[\"children\"][i],target,current\_depth + 1 ,max\_depth)\n",

" \n",

" if result is not None:\n",

" return result,True\n",

" \n",

" bottom\_reached = bottom\_reached and bottom\_reached\_rec \n",

" \n",

" return None,bottom\_reached\n",

" \n",

" \n",

"start={\n",

" \"value\":0,\"children\":[\n",

" {\" value \": 1, \"children\":[\n",

" {\" value \": 3, \"children\":[]},\n",

" {\" value \": 4, \"children\":[]}\n",

" ]},{\n",

" \" value \": 2, \"children\":[\n",

" {\" value \": 5, \"children\":[]},\n",

" {\" value \": 6, \"children\":[]}\n",

" \n",

" ]\n",

" \n",

" }\n",

" \n",

" ]\n",

"}\n",

"\n",

"print(iterative(start,6)[\"value\"])"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "753f8bbd",

"metadata": {},

"outputs": [],

"source": []

}

],

"metadata": {

"kernelspec": {

"display\_name": "Python 3.7.0 64-bit",

"language": "python",

"name": "python3"

},

"language\_info": {

"codemirror\_mode": {

"name": "ipython",

"version": 3

},

"file\_extension": ".py",

"mimetype": "text/x-python",

"name": "python",

"nbconvert\_exporter": "python",

"pygments\_lexer": "ipython3",

"version": "3.7.0"

},

"vscode": {

"interpreter": {

"hash": "999ea782e2d719ec62688e738a2ff20f2535cd73f1388dd13a2d835295a4fc1a"

}

}

},

"nbformat": 4,

"nbformat\_minor": 5

}